

Annals of Computer Science and Information Systems
Volume 7

Proceedings of the LQMR 2015 Workshop

September 13–16, 2015. Łódź, Poland



Tomasz Lechowski, Przemysław Wałęga,
Michał Zawidzki (eds.)



Annals of Computer Science and Information Systems, Volume 7

Series editors:

Maria Ganzha,

Systems Research Institute Polish Academy of Sciences and Warsaw University of Technology, Poland

Leszek Maciaszek,

Wrocław University of Economy, Poland and Macquarie University, Australia

Marcin Paprzycki,

Systems Research Institute Polish Academy of Sciences and Management Academy, Poland

Senior Editorial Board:

Wil van der Aalst,

Department of Mathematics & Computer Science, Technische Universiteit Eindhoven (TU/e), Eindhoven, Netherlands

Marco Aiello,

Faculty of Mathematics and Natural Sciences, Distributed Systems, University of Groningen, Groningen, Netherlands

Mohammed Atiquzzaman,

School of Computer Science, University of Oklahoma, Norman, USA

Barrett Bryant,

Department of Computer Science and Engineering, University of North Texas, Denton, USA

Ana Fred,

Department of Electrical and Computer Engineering, Instituto Superior Técnico (IST—Technical University of Lisbon), Lisbon, Portugal

Janusz Górski,

Department of Software Engineering, Gdansk University of Technology, Gdansk, Poland

Mike Hinchey,

Lero—the Irish Software Engineering Research Centre, University of Limerick, Ireland

Janusz Kacprzyk,

Systems Research Institute, Polish Academy of Sciences, Warsaw, Poland

Irwin King,

The Chinese University of Hong Kong, Hong Kong

Juliusz L. Kulikowski,

Natęcz Institute of Biocybernetics and Biomedical Engineering, Polish Academy of Sciences, Warsaw, Poland

Michael Luck,

Department of Informatics, King's College London, London, United Kingdom

Jan Madey,

Faculty of Mathematics, Informatics and Mechanics at the University of Warsaw, Warsaw, Poland

Andrzej Skowron,

Faculty of Mathematics, Informatics and Mechanics at the University of Warsaw, Warsaw, Poland

Editorial Associate: Katarzyna Wasielewska,

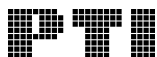
Systems Research Institute Polish Academy of Sciences, Poland

TeXnical editor: Aleksander Denisiuk,

University of Warmia and Mazury in Olsztyn, Poland

Proceedings of the LQMR 2015 Workshop

Tomasz Lechowski, Przemysław Wałęga,
Michał Zawidzki (eds.)



2015, Warszawa,
Polskie Towarzystwo
Informatyczne

Annals of Computer Science and Information Systems, Volume 7
Proceedings of the E2LP 2015 Workshop

USB: ISBN 978-83-60810-79-8

WEB: ISBN 978-83-60810-78-1

ISSN 2300-5963

DOI 10.15439/978-83-60810-78-1

© 2015, Polskie Towarzystwo Informatyczne
Al. Solidarności 82A m. 5
01-003 Warsaw
Poland

Contact: secretariat@fedcsis.org
<http://annals-csis.org/>

Cover:

Jana Waleria Denisiuk,
Elbląg, Poland

Also in this series:

Volume 6: Position Papers of the 2015 Federated Conference on Computer Science and Information Systems, **ISBN WEB: 978-83-60810-76-7, ISBN USB: 978-83-60810-77-4**

Volume 5: Proceedings of the 2015 Federated Conference on Computer Science and Information Systems, **ISBN WEB: 978-83-60810-66-8, ISBN USB: 978-83-60810-67-5**

Volume 4: Proceedings of the E2LP Workshop, **ISBN WEB: 978-83-60810-64-4, ISBN USB: 978-83-60810-63-7**

Volume 3: Position Papers of the 2014 Federated Conference on Computer Science and Information Systems, **ISBN WEB: 978-83-60810-60-6, ISBN USB: 978-83-60810-59-0**

Volume 2: Proceedings of the 2014 Federated Conference on Computer Science and Information Systems, **WEB: ISBN 978-83-60810-58-3, USB: ISBN 978-83-60810-57-6,**

ART: ISBN 978-83-60810-61-3

Volume 1: Position Papers of the 2013 Federated Conference on Computer Science and Information Systems (FedCSIS), **ISBN WEB: 978-83-60810-55-2, ISBN USB: 978-83-60810-56-9**

WELCOME to the LQMR Workshop on Logics for Qualitative Modelling and Reasoning. It is our great pleasure and honour to hold LQMR Workshop collocated with Federated Conference on Computer Science and Information Systems (FedCSIS) as a part of the Advances in Artificial Intelligence and Applications (AAIA) thematic area, taking place in Łódź, Poland, on September 13, 2015. On behalf of the LQMR Workshop Organizing Committee we cordially welcome all participants.

The idea of organizing this workshop originated from the project *Logics for Qualitative Reasoning* funded by the National Science Centre (DEC-2011/02/A/HS1/00395). The project is concerned with the logical foundations of qualitative representation and reasoning applied in artificial intelligence. Qualitative Reasoning (QR) has emerged as a sub-field of Artificial Intelligence to deal with representation and reasoning about continuous aspects of entities and systems in a symbolic, but human-like manner. The main issue in the QR approach is to develop an adequate tool for modeling situations in which information is not sufficiently precise or cannot be described by numerical values.

The project aims to develop logical theories and tools for qualitative representation and reasoning, with applications to many domains for which qualitative inference methods are significant.

The project is also aimed at analysis of model-theoretic properties of qualitative logics, such as definability and expressive power, finite model property, and decidability, among others. The third research objective is the construction and implementation of deduction systems for the logics developed in the project. We focus on decidable logics and their automated decision procedures in the style of relational dual tableaux.

The LQMR workshop categorically addresses the theory and application of logical formalisations of qualitative reasoning within engineering, technical, and computational

cognitive systems. The workshops will build bridges between different research groups interested in qualitative modelling and reasoning. In particular, perspectives from logic and computer science employing formal methods for QR, formal methods for spatial reasoning, and researchers dealing with fundamental philosophical aspects of QR are of focus. Additionally, problems of more applied nature in the field of engineering and artificial intelligence are also emphasised.

The contributed papers focus on three main areas: qualitative spatial reasoning, its possible applications, and applications of qualitative methods to philosophical problems.

In addition to the contributed papers, four invited keynote talks were delivered: by prof. Thomas Bittner from State University of New York at Buffalo, who spoke on vague region-based geometry, by dr Ian Pratt-Hartmann from The University of Manchester, who devoted his presentation to topological logics of Euclidean spaces, by prof. Kenneth Forbus, who spoke on three frontiers for qualitative reasoning, and by prof. Ivan Bratko, whose lecture was concerned with the problem of learning qualitative models.

These Proceedings will augment state of the art in Qualitative Reasoning with several excellent references.

We thank all authors and participants for their contributions.

LQMR Workshop Co-Chairs:

Tomasz Lechowski, Institute of Philosophy, University of Warsaw, Poland

Przemysław Wałęga, Institute of Philosophy, University of Warsaw, Poland

Michał Zawidzki, Department of Logic, University of Łódź, Institute of Philosophy, University of Warsaw, Poland

Proceedings of the LQMR Workshop

September 13–16, 2015. Łódź, Poland

TABLE OF CONTENTS

10TH INTERNATIONAL SYMPOSIUM ADVANCES IN ARTIFICIAL INTELLIGENCE AND APPLICATIONS

Call For Papers	1
-----------------	---

1ST WORKSHOP ON LOGICS FOR QUALITATIVE MODELLING AND REASONING

A Qualitative Model for Reasoning about 3D Objects using Depth and Different Perspectives <i>Zoe Falomir</i>	3
Spatial Rules for Capturing Qualitatively Equivalent Configurations in Sketch maps <i>Sahib Jan, Carl Schultz, Angela Schwering, Malumbo Chipofya</i>	13
A Framework for Constructing Correct Qualitative Representations of Geometries using Mereology over Bintrees <i>Leif Harald Karlsen, Martin Giese</i>	21
On (in)Validity of Aristotle's Syllogisms Relying on Rough Sets <i>Tamás Kádek, Tamás Mihálydeák</i>	35
Parthood and Convexity as the Basic Notions of a Theory of Space <i>Klaus Robering</i>	41
Encoding Relative Orientation and Mereotopology Relations with Geometric Constraints in CLP(QS) <i>Carl Schultz, Mehul Bhatt</i>	55
Using Mathematical Modeling as an Example of Qualitative Reasoning in Metaphysics. A Note on a Defense of the Theory of Ideas <i>Bartłomiej Skowron</i>	65

10th International Symposium Advances in Artificial Intelligence and Applications

THE AAIA'15 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'15 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'15 will provide a forum for the exchange of ideas between theoreticians and practitioners to address the important issues.

TOPICS

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 Systems
- Approximate Reasoning
- Fuzzy Modeling and Control
- Data Mining
- Web Mining
- Machine Learning
- Combining Multiple Knowledge Sources in an Integrated Intelligent System
- Neural Networks
- Evolutionary Computation
- Nature Inspired Methods
- Natural Language Processing
- Image Processing and Interpreting
- Applications in Bioinformatics
- Hybrid Intelligent Systems
- Granular Computing
- Architectures of Intelligent Systems
- Robotics
- Real-world Applications of Intelligent Systems
- Rough Sets

PROFESSOR ZDZISLAW PAWLAK BEST PAPER AWARDS

We are proud to announce that we will continue the tradition started during the AAIA'06 Symposium and award two "Professor Zdzislaw Pawlak Best Paper Awards" for contributions which are outstanding in their scientific quality. The two award categories are:

- Best Student Paper - for graduate or PhD students. Papers qualifying for this award must be marked as "Student full paper" to be eligible for consideration.
- Best Paper Award for the authors of the best paper appearing at the Symposium.

Candidates for the awards can come from AAiA and all workshops organized within its framework (i.e. AIMaViG, AIMA, ASIR, CEIM, LQMR, WCO).

In addition to a certificate, each award carries a prize of 300 EUR provided by the Mazowsze Chapter of the Polish Information Processing Society.

IFSA AWARD FOR YOUNG SCIENTIST

During the Advances in Artificial Intelligence and Applications (AAIA) Symposium, the International Fuzzy Systems Association (IFSA) Best Paper Award for Young Scientist, will be presented.

Candidates for the awards can come from AAiA and all workshops organized within its framework (i.e. AIMaViG, AIMA, ASIR, CEIM, LQMR, WCO).

EVENT CHAIRS

Janusz, Andrzej, University of Warsaw, Poland
Ślęzak, Dominik, University of Warsaw & Infobright Inc., Poland
Event Chairs

ADVISORY BOARD

Kacprzyk, Janusz, Systems Research Institute, Warsaw, Poland
Kwaśnicka, Halina, Wrocław University of Technology, Poland
Markowska-Kaczmar, Urszula, Wrocław University of Technology, Poland
Skowron, Andrzej, University of Warsaw, Poland

PROGRAM COMMITTEE

Artiemjew, Piotr, University of Warmia and Mazury, Poland
Bartkowiak, Anna, Wrocław University, Poland
Bazan, Jan, University of Rzeszów, Poland
Bodyanskiy, Yevgeniy, Kharkiv National University of Radio Electronics, Ukraine
Błaszczyszki, Jerzy, Poznań University of Technology, Poland
Cetnarowicz, Krzysztof, AGH University of Science and Technology, Poland
Chakraverty, Shampa, Netaji Subhas Institute of Technology, India
Cheung, William, Hong Kong Baptist University, Hong Kong S.A.R., China
Cyganek, Bogusław, AGH University of Science and Technology, Poland
Czarnowski, Ireneusz, Gdynia Maritime University, Poland
Dardzinska, Agnieszka, Białystok University of Technology, Poland
Dey, Lipika, Tata Consulting Services, India
Duentsch, Ivo, Computer Science Department, Brock University, Canada
Froelich, Wojciech, Institute of Computer Science, University of Silesia, Poland
Girardi, Rosario, Federal University of Maranhão, Brazil
Hassanien, Aboul Ella, Cairo University, Egypt
Herrera, Francisco, University of Granada, Spain
Holzinger, Andreas, Graz University of Technology, Austria

Jaromeczyk, Jerzy W., University of Kentucky, United States
Jin, Xiaolong, Chinese Academy of Sciences, China
Jin, Peng, Leshan Normal University, China
Kayakutlu, Gulgun, Istanbul Technical University, Turkey
Korbicz, Józef, University of Zielona Gora, Poland
Krasuski, Adam, The Main School of Fire Service (SGSP), Poland
Kuznetsov, Sergei, National Research University - Higher School of Economics, Russia
Lewis, Rory, University of Colorado at Colorado Springs, United States
Loukanova, Roussanka, Department of Mathematics, Stockholm University, Sweden
Marek, Victor, University of Kentucky, United States
Matson, Eric T., Purdue University, United States
Menasalvas, Ernestina, Universidad Politécnica de Madrid, Spain
Mercier-Laurent, Eunika, IAE Lyon3, France
Mihálydeák, Tamás, University of Debrecen, Hungary
Miroslaw, Lukasz, University of Applied Science Rapperswil & Wrocław University of Technology, Switzerland
Miyamoto, Sadaaki, University of Tsukuba, Japan
Moshkov, Mikhail, King Abdullah University of Science and Technology, Saudi Arabia
Myszkowski, Pawel, Wrocław University of Technology, Poland
Ngan, Ben C. K., The Pennsylvania State University, United States
Nourani, Cyrus F., Akdmkrd-DAI TU Berlin, CBS Copenhagen-TansMedia GmbH, Munich, and SFU Burnaby, Canada
Nowostawski, Mariusz, Gjøvik University College, Norway
Pancerz, Krzysztof, University of Management and Administration in Zamość, Poland
Paradowski, Mariusz, Wrocław University of Technology, Poland
Peters, Georg, Munich University of Applied Sciences, Germany
Porta, Marco, University of Pavia, Italy
Przybyła-Kasperek, Małgorzata, University of Silesia, Poland

Ramanna, Sheela, University of Winnipeg, Canada
Ras, Zbigniew, University of North Carolina at Charlotte, United States
Reformat, Marek, University of Alberta, Canada
Santos Jr., Eugene, Dartmouth College, United States
Sas, Jerzy, Wrocław University of Technology, Poland
Schaefer, Gerald, Loughborough University, United Kingdom
Sikora, Marek, Silesian University of Technology, Poland
Snasel, Vaclav, VSB -Technical University of Ostrava, Czech Republic
Sydow, Marcin, Polish Academy of Sciences and Polish-Japanese Acad. of IT, Poland
Szczęch, Izabela, Poznań University of Technology, Poland
Szpakowicz, Stan, University of Ottawa, Canada
Szwed, Piotr, AGH University of Science and Technology, Poland
Tsay, Li-Shiang, North Carolina A&T State University, United States
Unland, Rainer, Universität Duisburg-Essen, Germany
Unold, Olgierd, Wrocław University of Technology, Poland
Wang, Xin, University of Calgary, Canada
Wieczorkowska, Alicja, Polish Japanese Academy of Information Technology, Poland
Wiśniewski, Piotr, Nicolaus Copernicus University, Poland
Wozniak, Michał, Wrocław University of Technology, Poland
Wysocki, Marian, Rzeszów University of Technology, Poland
Zadrozny, Sławomir, Systems Research Institute, Poland
Zaharie, Daniela, West University of Timisoara, Romania
Zakrzewska, Danuta, Łódź University of Technology, Poland
Zielosko, Beata, University of Silesia, Poland
Zighed, Djamel Abdelkader, University of Lyon, Lyon 2, France
Ziolko, Bartosz, AGH University of Science and Technology, Poland

A Qualitative Model for Reasoning about 3D Objects using Depth and Different Perspectives

Zoe Falomir

Cognitive Systems (CoSy) Department

Spatial Cognition Centre

Universität Bremen

Enrique-Schmidt-Str. 5, 28359 Bremen, Germany

Email: zfalomir@informatik.uni-bremen.de

Abstract—A qualitative model for describing 3D objects (*Q3D*) using depth and different perspectives is presented in this paper. The front, right and up perspectives are considered as canonical. The *Q3D* model allow reasoning through logics defined to test the consistency of descriptions. The maximal volume of the object is also obtained logically using its *Q3D* description. Moreover, this model infers some features of the unknown perspectives of the object by defining logics based on the continuity of holes and the relative depth presented by opposite perspectives. The *Q3D* logics are implemented in Prolog and promising results are obtained, which can inspire approaches to solve 3D spatial problems computationally.

I. INTRODUCTION

QUALITATIVE Spatial and Temporal Representations and Reasoning (QSTR) [1]–[3] models and reasons about *time* (i.e. coincidence, order, concurrency, overlap, granularity) and also about properties of *space* (i.e. topology, location, direction, proximity, geometry, intersection, etc.) and their evolution between continuous neighbouring situations. Maintaining the consistency in space and time are the basics in qualitative reasoning when solving spatial and temporal problems. Spatio-temporal reasoning models deal with imprecise and incomplete knowledge on a symbolic level and have been successful in many areas and applications such as robotics [4], [5], computer vision [6], [7], ambient intelligence [8], [9], 2D shape description and recognition [10], colour naming and similarity [11], architecture and design [12], spatial query solving in geographic information systems [13], [14], etc. Furthermore, qualitative representations are thought to be closer to the cognitive domain, as shown in cognitive models of sketch recognition [15], spatial problem solving tasks (i.e. visual oddity tasks) [16]. However, further research is still needed to combine more aspects of QSTR with cognitive spatial thinking.

In the fields of computer vision, robotics and ambient intelligence, 3D object description and recognition are challenging tasks nowadays. Dealing with three dimensional data is a challenge because they usually suffer from distortions due to noisy sensors, viewpoint changes and point density variations. In the computer vision literature, approaches for object recognition usually use 3D descriptors to encode their shapes from different perspectives [17], [18]: feature-based approaches describe the local or global properties of the surface of the object (i.e., colour, curvature, texture, etc.); graph-based approaches describe the structure or skeleton of the object, that is, the relations between the object parts; and other approaches use other techniques like extended gaussian images, 3D moments, volumetric errors, etc.

Research in the field of 3D object recognition has been fostered by the availability of low-cost depth cameras based on structured infrared light (also called RGB-Depth cameras) such as the Microsoft Kinect and the Asus Xtion¹. Since the development of these sensors, diverse techniques have appeared to recognise real objects which learn their shape from the thousands of points which describe their surface from different perspectives [19]–[21]. Although these techniques are successful and applied in robotics and ambient intelligent systems, they are quite computational expensive, and they are not exploiting constraints in space to reduce this cost.

In the field of psychology, spatial cognition studies have demonstrated that there is a strong link between success in Science, Technology, Engineering and Math (STEM) disciplines and spatial abilities [22], [23]. Thus, it is important to maintain and train these abilities from the early stages. For example, children at 4 years old have already informal awareness of spatial relations such as parallel relations for two dimensional shape identification before they are properly taught about parallelism [24]. For this reason, researchers in US and Canada study the actualities and possibilities of training/including spatial reasoning in contemporary school mathematics [25], also because spatial learning and reasoning can be taught easily using visual and kinetic interactions offered by new digital technologies [26]. For example, touchscreen digital devices can facilitate geometrical expression for young children [27]. High spatial skills are also required in space teleoperation [28] (mental rotation and perspective-taking strategies are proved to be used by the operator-astronaut to move a robot arm around the workspace) and they are also decisive in Medicine [29].

Moreover, in cognitive psychology, games like *Upside Down World* are used to evaluate students' spatial skills when they are challenged to recreate buildings composed of multilink cubes and to use spatial language to describe the composition of these buildings so that their colleagues can build accordingly [25]. A test of the German Academic Foundation to find children with gifted brains among candidates for scholarships consists in finding out the consistent view/projection for a 3D object usually corresponding to a technological drawing².

This paper explores the challenge of describing 3D objects qualitatively and it is based on the levels of depth each object

¹Trade and company names are included for benefit of the reader and imply no endorsement or preferential treatment of the product by the author.

²Test der Studienstiftung: Gehirnjogging für Hochbegabte, see Spiegel Online: <http://www.spiegel.de/quiztool/quiztool.249771.html>

has at each perspective. This approach is inspired in designs of pieces which abstract the main features of the object from all their properties in the real world and describe them using 3 canonical views (top, lateral and front). Moreover, this approach is cognitively based, since in experimental psychology there is support for the general idea that human object recognition involves view-dependent representations, that is, people prefer to imagine, view, or photograph objects from certain “canonical” views [30]. Also this approach has been motivated by the fact that the German Academic Foundation uses consistent view/projection of a 3D object corresponding to a technological drawing to measure intelligence in humans². An example of a question in this test and the instructions given may be that provided in Fig. 1. Note that this example is made up for this paper to avoid copyright issues, and that real examples can be obtained online².

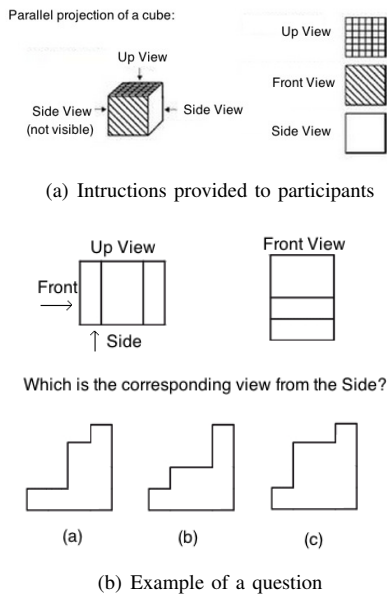


Fig. 1. (a) Instructions of the test translated to English; (b) Plausible example of a question regarding 3D projections in the German Academic Foundation test.

The rest of the paper is organised as follows. In Section II, properties of spatial substrates are explained. Section III presents a model for Qualitative Description of 3D objects. Section IV explains the logics encountered, and the consistency conditions for the *Q3D* model are described in Section V. Section VI presents a logic approach to obtain the maximal volume of an object described by the *Q3D* model. Section VII explains how to infer some features of the occluded views (back, down and left) from their opposite views (front, up and right). In Section VIII, the implementation of the model is described. Section IX discusses the closer related work. And conclusions and future work are presented in Section X.

II. SPATIAL SUBSTRATES AND THEIR PROPERTIES

As Freksa [31] mentions, *properties of spatial objects and configurations are intrinsically highly interdependent. If we modify one spatial aspect (e.g. distance, orientation, topological relation) in a spatial structure, other spatial aspects will be*

changed automatically, as well. We call such a structure a spatial substrate. If we move an object in space, the spatial locations of all its parts as well as their relations to other objects will change. If we change a single spatial aspect in a spatial substrate, all these changes take place (for free); no computing (or otherwise) effort is required.

As far as we are concerned, there is no related literature about which are the properties spatial substrates may have. Here, we appeal to the intuition of the reader to formulate some properties which we envision they help in solving spatial problems:

- **Abstraction:** people abstract dimensions in space (i.e., by assuming one dimension as constant) and re-represent data in a way that helps visualising a problem. For example, a map represents 3D space in a 2D paper, sometimes assuming relief or altitude as constant.
- **Continuity:** dimensions in space are continuous. They can be abstracted or considered as constant in a representation, but this representation must be coherent with the space and transmit changes in the dimension abstracted, if produced. For example, if a change in relief is produced (i.e., a road is cut) this change should be transmitted to the dimensions not abstracted (i.e., an interactive or up-to-date map should represent this discontinuity in the road).
- **Interrelation:** most dimensions in space are relative or interrelated to each other. For example, when comparing roads in a map, people usually look for the shorter-path (wrt another) or the quicker path (wrt another). If the roads are represented by abstracting the same dimension, then they can be compared directly. If one road considers relief while the other does not, then they are not comparable.

In 3D engineering object design (see Fig. 2), objects are usually **abstracted or re-represented** using 3 canonical views. In each view, the object is abstracted by considering a dimension as constant. For example, in the *front view/perspective*, the dimensions involved are the width and height of the object, while the depth dimension is assumed as constant; in the *right view*, the dimensions depth and height are represented, whereas the width dimension is assumed as constant; and in the *up view*, the dimensions represented are width and depth, while height is assumed as constant.

Note that, in contrast to the 3D projection test by the German Academic Foundation where views are provided disconnected from each other, in 3D technological drawings, engineers assume **continuity** in their abstractions or re-representations of the object. When assuming a constant value for a dimension, it is assumed also that this dimension is continuous. If a change is produced in the dimension abstracted, this change has to be reflected in the other representations. In Fig. 2 this continuity is represented as grey lines.

Moreover, as Fig. 2 shows, in 3D technological drawings, perspectives are **relative** to each other. For example, the dimension height is involved in the views front and right; similarly, the dimension depth is involved in the views right and up; and the dimension width is involved in the views up and front. Therefore, following the continuity principle, a change in each common dimension must be reflected in the other two views involved.

After observing these properties in the spatial substrates, the following model for qualitative 3D object description was defined.

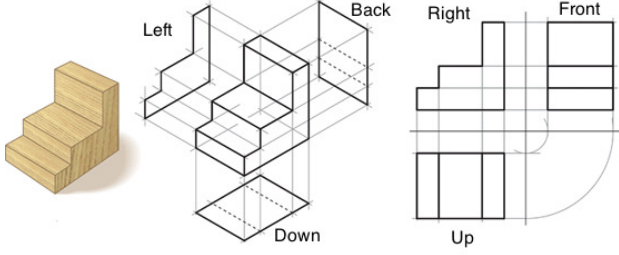


Fig. 2. Example of a 3D object in a technological drawing which shows the corresponding relationships among perspectives.

III. A QUALITATIVE DESCRIPTOR FOR 3D OBJECTS

When thinking qualitatively about real objects in space, humans usually think about volumes. For example, in pictures and paintings, observers assume sometimes depth in objects/scenes –differentiating foreground from background [11]– which is not easy to see in the absence of shadows. As a consequence, the minimal unit for the qualitative description presented here is considered a volume, specifically a cube of side $x \in \mathbb{R}$, which may be used to build an object similarly to how pixels are used to build digital images.

Therefore, a reference system for qualitative 3D object description is defined as follows:

$$Q3D_{RS} = \{F, R, U \in P \mid P \subseteq N_{depths}\}$$

$$N_{depths} = \{a, b, c, d, \dots, *\}$$

where F, R and U are the Front, Right and Up perspectives (P) or views of the object, and N is the total number of cubes which compose each edge of the object. That is, the edges of the object in each perspective are described by the volume of cubes of equal size, being the basic unit of measure considered a cube of side $x \in \mathbb{R}$ (i.e., $x = 1\text{cm}$, $x = 0.75\text{cm}$, $x = 5\text{m}$, etc.).

Thus, each perspective has N levels of depth, which can be named differently and sequentially as $\{a, b, c, d, \dots, *\}$ where a is the surface of the cube, b is the first level of depth (a previous cube in the row has been removed), c is the second level of depth (two previous cubes in the row have been removed) and so on, until $*$ is reached, which indicates that all the cubes in a row have been removed. The description is started from the upper-left part at each perspective.

As a first example, let us consider the object in Fig. 3 and its corresponding description according to the views: Front (F) in red, Right (R) in blue, and Up (U) in yellow. Starting from the

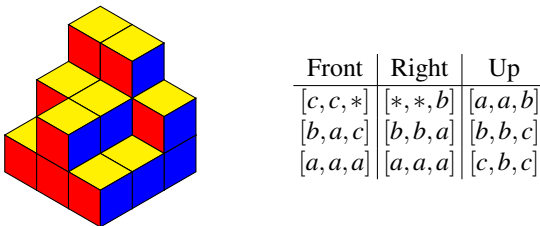
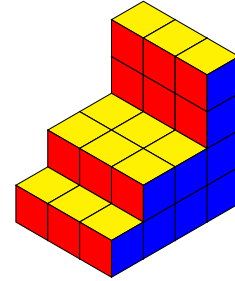


Fig. 3. Example of 3D object divided by a $3 \times 3 \times 3$ grid of cubes showing the front (red), right (blue) and up (yellow) views, and its corresponding Q3D description.

upper-left part of the *front* perspective, it can be observed that

2 cubes were removed in the first row, and also in the second row, so this is represented by the parameters c, c in the Q3D description. Then, all the cubes have been removed in the third row, so this is represented by the parameter $*$. Going down a level, it can be observed that only one cube is left in the first row (represented by b), then all the cubes are filling the second row (represented as a) and, in the third row, two cubes are missing (represented as c). Finally, in the basis of the object, all the rows are complete, which is represented as a, a, a . The perspectives *right* and *up* are explained similarly.

As a second example, let us consider the object in Fig. 4 extracted from the technological drawing in Fig. 2. The proportions of the object show that it can be modelled by a grid of $4 \times 4 \times 3$ cubes to be described qualitatively according to the different levels of depth at each perspective. Fig. 4 shows its corresponding Q3D description according to all the possible views.



Right	Front	Up	Left	Back	Down
$[*, *, *, a]$	$[d, d, d]$	$[d, c, c, a]$	$[a, *, *, *]$	$[a, a, a]$	$[a, a, a, a]$
$[*, *, *, a]$	$[d, d, d]$	$[d, c, c, a]$	$[a, *, *, *]$	$[a, a, a]$	$[a, a, a, a]$
$[*, a, a, a]$	$[b, b, b]$	$[d, c, c, a]$	$[a, a, a, *]$	$[a, a, a]$	$[a, a, a, a]$
$[a, a, a, a]$	$[a, a, a]$		$[a, a, a, a]$	$[a, a, a]$	

Fig. 4. Three dimensional object representation extracted from the technological drawing in Fig. 2 which can be divided into a $4 \times 4 \times 3$ grid of cubes to be described qualitatively by the Q3D approach.

It is important to notice that a change in a parameter or letter in the Q3D description in Fig. 4 corresponds to a line of the sketch drawing in Fig. 2. This is easily seen in the *up perspective* descriptor, where a line may be drawn vertically separating all d letters from c letters and another line may be drawn vertically separating c letters from a letters (since each different letter correspond to a different depth), so that the technological drawing related to up perspective shown in Fig. 2 would be obtained. Therefore, some hints about the **shape** of the object are obtained. However, note that the complete shape of the object is not described at this stage, and also that circular or squared holes in an object would be represented equally by $*$, described by the change in depth they produce.

IV. Q3D LOGICS FOR DESCRIBING OBJECTS

The Q3D description of an object can be also described logically, as follows:

$$\forall X \text{ } Q3DObject(X) \rightarrow \text{view}(\text{front}, X, N, N', Q3D) \wedge \text{view}(\text{right}, X, N, N', Q3D) \wedge \text{view}(\text{up}, X, N, N', Q3D) \quad (1)$$

where X is a particular object; N is the dimension in cubes of the edge of the object; and $Q3D$ is the qualitative description corresponding to each of the perspectives *front*, *right* and *up*, which is built by N lists of N' elements of depth each.

The Q3D logic description for the object in Fig. 3 is provided as follows:

$$\begin{aligned} Q3DObject(object_1) \rightarrow \\ & view(front, object_1, 3, 3, [[c, c, *], [b, a, c], [a, a, a]]) \wedge \\ & view(right, object_1, 3, 3, [[*, *, b], [b, b, a], [a, a, a]]) \wedge \\ & view(up, object_1, 3, 3, [[a, a, b], [b, b, c], [c, b, c]]) \end{aligned} \quad (2)$$

The Q3D logic description for the object in Fig. 4 is provided as follows:

$$\begin{aligned} Q3DObject(object_2) \rightarrow \\ & view(front, object_2, 4, 3, \\ & \quad [[d, d, d], [d, d, d], [b, b, b], [a, a, a]]) \wedge \\ & view(right, object_2, 4, 4, \\ & \quad [[*, *, *, a], [*, *, *, a], [*, a, a, a], [a, a, a, a]]) \wedge \\ & view(up, object_2, 3, 4, \\ & \quad [[d, c, c, a], [d, c, c, a], [d, c, c, a]]) \end{aligned} \quad (3)$$

V. REASONING WITH THE Q3D: CONSISTENT AND INCONSISTENT PERSPECTIVES

According to spatial reasoning, from the perspectives Front (F), Right (R) and Up (U), an object can be built in a three-dimensional space. In mechanical engineering, it is assumed as a convention that this canonical views correspond to the more detailed views. So, which are the common sense facts in spatial reasoning which guide this building? What are the 3D spatial facts which can or cannot happen?

Let us consider the representation in Fig. 5 to exemplify the following cases:

- Case 1: a change in an edge affects 2 perspectives at least. For example, if the cube $\{F_{1,2}, U_{3,2}\}$ disappears, this must be reflected at both perspectives F and U.
- Case 2: a change in a vertex affects 3 perspectives. For example, if the cube $\{F_{1,3}, R_{1,1}, U_{3,3}\}$ disappears, this must be reflected at perspectives Front and Up, but also at Right.
- Case 3: each hole affects 2 perspectives at least, two of them corresponding to opposite views. For example, a hole in the middle of the object (i.e., cube $\{F_{2,2}\}$ and followers disappear) would affect Front and Back perspectives, whereas a hole involving cubes $\{F_{2,3}, R_{2,1}, R_{2,2}, R_{2,3}\}$ would affect 3 perspectives: Front, Right and Back.

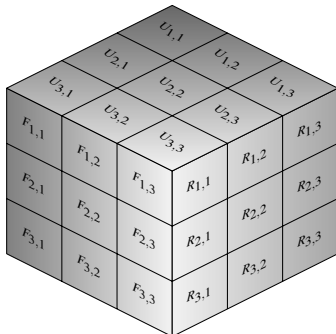


Fig. 5. Example of an object showing the constraints at the boundary of the canonical perspectives.

The spatial constraints appear along the boundary of the perspectives or the edges of the object, since a change in a perspective must be consistent with a change in another

perspective. In Fig. 5, each cube is named according to the perspectives Front (F), Right (R) and Up (U). Therefore, the descriptions must be consistent where the edges meet at F-R, F-U, and R-U perspectives:

$$\begin{aligned} consistent_Q3D(F, R, U) \rightarrow \\ & consistent_perspective(F, R) \wedge \\ & consistent_perspective(F, U) \wedge \\ & consistent_perspective(R, U) \end{aligned} \quad (4)$$

Note that the problem is simplified by abstracting one dimension/view in each comparison, that is, the views meeting at each edge are those related and those that must be consistent.

Let us consider the edges meeting at cube $\{F_{1,3}, R_{1,1}, U_{3,3}\}$, then the consistent conditions for front (F) and right (R) perspectives can be defined as:

$$\begin{aligned} consistent_perspective([F_{11}, F_{12}, F_{13}], [F_{21}, F_{22}, F_{23}], [F_{31}, F_{32}, F_{33}], \\ [R_{11}, R_{12}, R_{13}], [R_{21}, R_{22}, R_{23}], [R_{31}, R_{32}, R_{33}]) \rightarrow \\ & consistent_side(F_{13}, [R_{11}, R_{12}, R_{13}]) \wedge \\ & consistent_side(F_{23}, [R_{21}, R_{22}, R_{23}]) \wedge \\ & consistent_side(F_{33}, [R_{31}, R_{32}, R_{33}]) \wedge \\ & consistent_side(R_{11}, [F_{11}, F_{12}, F_{13}]) \wedge \\ & consistent_side(R_{21}, [F_{21}, F_{22}, F_{23}]) \wedge \\ & consistent_side(R_{31}, [F_{31}, F_{32}, F_{33}]) \end{aligned} \quad (5)$$

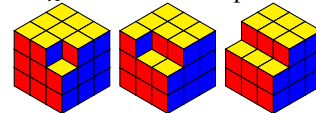
The conditions to obtain a consistent perspective in the sides R-U and F-U are defined similarly. It has been observed that the same constraints must be fulfilled for each edge, F-R, R-U and F-U, so they can be generalised:

$$\begin{aligned} consistent_side(F_{i3}, [R_{i1}, R_{i2}, R_{i3}]) \rightarrow \\ & level_a(F_{i3}) \wedge level_a(R_{i1}) \end{aligned} \quad (6)$$

The logic rule (6) is explained as follows: if any cube exists on the right edge at *front* perspective, then it must exist also a cube on the left edge at *right* perspective, since a cube involves a volume which continues in both dimensions or perspectives. Note that *i* means *some row* and 1,2,3 means *column 1,2,3*, respectively. Note also that nothing is constrained on the R_{i2} and R_{i3} cubes.

$$\begin{aligned} consistent_side(F_{i3}, [R_{i1}, R_{i2}, R_{i3}]) \rightarrow \\ & level_b(F_{i3}) \wedge level_a(R_{i2}) \wedge \\ & (level_b(R_{i1}) \vee level_c(R_{i1}) \vee no_exist(R_{i1})) \end{aligned} \quad (7)$$

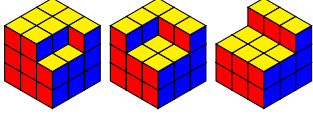
The logic rule (7) is explained as follows: if only a cube disappear on the right edge at *front* perspective (that is, $F_{i3} \equiv b$ in Q3D), then it must exist also a cube located on the second column at *right* perspective, since this is the cube seen from the *front* (that is, $R_{i2} \equiv a$). And the constraints on the cubes located on the left edge at *right* perspective (R_{i1}) are that: they cannot be level *a* of depth since that would mean that missing cubes, appeared again, which is inconsistent. For R_{i1} all the other possibilities can happen:



Note that nothing is constrained on the R_{i3} cubes.

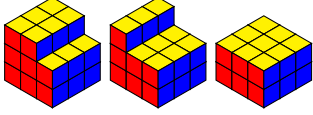
$$\begin{aligned} \text{consistent_side}(F_{i3}, [R_{i1}, R_{i2}, R_{i3}]) \rightarrow \\ \text{level_c}(F_{i3}) \wedge \text{level_a}(R_{i3}) \wedge \\ (\text{level_b}(R_{i1}) \vee \text{level_c}(R_{i1}) \vee \text{no_exist}(R_{i1})) \wedge \\ (\text{level_b}(R_{i2}) \vee \text{level_c}(R_{i2}) \vee \text{no_exist}(R_{i2})) \end{aligned} \quad (8)$$

The logic rule (8) is explained as follows: if two cubes disappear on the right edge at *front* perspective (that is, $F_{i3} \equiv c$ in Q3D), then it must exist also a cube located at the third column at *right* perspective, since this is the cube seen from the *front* (that is, $R_{i3} \equiv a$). And the constraints on the cubes located on the left edge at *right* perspective (R_{i1} and R_{i2}) are that: they cannot be level *a* of depth since that would mean that the missing cubes appeared again, and that is inconsistent. For R_{i1} and R_{i2} all the other possibilities can happen:



$$\begin{aligned} \text{consistent_side}(F_{i3}, [R_{i1}, R_{i2}, R_{i3}]) \rightarrow \\ \text{no_exist}(F_{i3}) \wedge \\ (\text{level_b}(R_{i1}) \vee \text{level_c}(R_{i1}) \vee \text{no_exist}(R_{i1})) \wedge \\ (\text{level_b}(R_{i2}) \vee \text{level_c}(R_{i2}) \vee \text{no_exist}(R_{i2})) \wedge \\ (\text{level_b}(R_{i3}) \vee \text{level_c}(R_{i3}) \vee \text{no_exist}(R_{i3})) \end{aligned} \quad (9)$$

The logic rule (9) can be explained as follows: if all the cubes disappear on the right edge at *front* perspective (that is, $F_{i3} \equiv *$ in Q3D), then no cube on the first row at right perspective must exist ($R_{i1} \neq a$, $R_{i2} \neq a$, $R_{i3} \neq a$) but all the rest of possibilities can happen for R_{i1} , R_{i2} and R_{i3} :

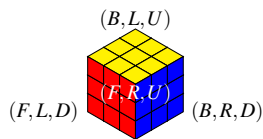


Note that F_{i3} denotes $F_{1,3}$, $F_{2,3}$ or $F_{3,3}$; that R_{i1} denotes $R_{1,1}$, $R_{1,2}$ or $R_{1,3}$; also R_{i2} denotes $R_{2,1}$, $R_{2,2}$ or $R_{2,3}$; and R_{i3} denotes $R_{3,1}$, $R_{3,2}$ or $R_{3,3}$, and also,

$$\begin{aligned} \forall X \text{ level_a}(X) &\rightarrow a \\ \forall X \text{ level_b}(X) &\rightarrow b \\ \forall X \text{ level_c}(X) &\rightarrow c \\ \forall X \text{ no_exist}(X) &\rightarrow * \end{aligned} \quad (10)$$

If information is given about the rest of perspectives (Back -B-, Left -L-, Down -D-), the consistency conditions between the edges at each perspective are defined similarly.

Note that, as each vertex is proving consistency in 3 edges, only by proving the consistency conditions in 4 opposite vertices in the cube, all the edges of the cube are covered. Let us show an example:



Proving the consistency at the 4 vertices in the drawing above ($\text{consistent_Q3D}(F, R, U)$, $\text{consistent_Q3D}(B, L, U)$, $\text{consistent_Q3D}(F, L, D)$, $\text{consistent_Q3D}(B, R, D)$) is enough to cover the 12 edges of a complete consistent description of a 3D object.

The computational complexity of the consistency algorithm is calculated as follows. When choosing 4 opposite vertices in the cube where to apply the *consistent_Q3D* function, all the 12 edges of the cube are covered, since the *consistent_Q3D* function checks the consistency of the 3 edges meeting at a specific vertex. Then, the final computational complexity is 12 times the complexity of the *consistent_perspective* function. And the complexity of this function is $2N$ where N is the edge size in volume-cubes. In summary, $12 \cdot 2 \cdot N = 24 \cdot N$, thus the computational cost is $\mathcal{O}(N)$.

VI. INFERRING THE MAXIMAL VOLUME OF THE OBJECT FROM THE Q3D

The maximal volume of an object described by a *Q3D* can be obtained as:

$$Q3D_{\text{volume}} = \min(\text{volume}_P(F), \text{volume}_P(R), \text{volume}_P(U)) \quad (11)$$

where $Q3D_{\text{volume}}$ refers to the volume of the object measured in cubes of side $x \in \mathbb{R}$; and \min refers to the minimum of the volumes corresponding to each perspective $F, R, U \in P$, that is (volume_P) which is defined as follows.

The volume of a perspective is the opposite to its levels of depth. If an object is described by N cubes, then the volume is calculated as:

$$\text{volume}_P(P) = \sum_{i=1}^{N \cdot N} \text{volume}(\sigma) \quad (12)$$

where, the volume of an element σ is defined as:

$$\text{volume}(\sigma) = \sum_{i=1}^{\sigma} N - i \quad (13)$$

that is, for example, for $N=3$, $\text{volume}(a) \equiv N$, $\text{volume}(b) \equiv N - 1$, $\text{volume}(c) \equiv N - 2$ and $\text{volume}(*) \equiv 0$.

As an example, the volume of the object in Fig. 3 is calculated as follows:

Object 1	Front view	Right view	Up view
if $N=3$,	$[c, c, *] = 1 + 1 + 0$	$[*, *, b] = 0 + 0 + 2$	$[a, a, b] = 3 + 3 + 2$
$a=3, b=2$	$[b, a, c] = 2 + 3 + 1$	$[b, b, a] = 2 + 2 + 3$	$[b, b, c] = 2 + 2 + 1$
$c=1, *=0$	$[a, a, a] = 3 + 3 + 3$	$[a, a, a] = 3 + 3 + 3$	$[c, b, c] = 1 + 2 + 1$
	= 17	= 18	= 17

Note that the minimal result obtained is 17, which is the correct volume of the object, as it can be checked in Fig. 3.

The minimum volume of all canonical perspectives is calculated, since, for example, some holes may only be seen from a specific perspective. Then, it is important to notice that the volume obtained is the maximal that the object can have when being observed from the FRU perspective. Note that it is important to select the canonical perspectives as FRU, otherwise the correct volume might not be obtained since an object might have a hole at a side which would not be appreciated.

The computational cost of the maximal volume algorithm is obtained as follows. The complexity of calculating the volume of a *Q3D* view is the complexity of getting the value of the N depths at each row and the N depths at each column, being N the size of the edge, thus the cost is $N \cdot N$. This volume must be computed using the 3 views at a vertex (i.e., FRU), that is $3 \cdot N \cdot N$, so the computational complexity of the algorithm is $\mathcal{O}(N^2)$.

VII. INFERRING SOME DEPTHS IN UNKNOWN PERSPECTIVES FROM OPPOSITE VIEWS

Taking into account the spatial relations showed by the drawing in Fig. 6, from the views Front (F), Right (R) and Up (Up), how much can we deduce logically from the rest of the object? Can the rest of the views be computed?

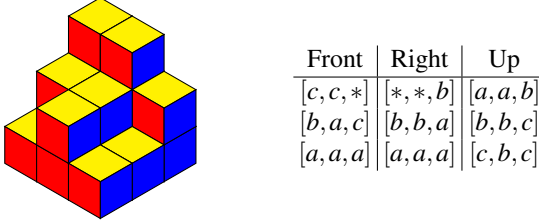


Fig. 6. Q3D description of object 1 used to explain inferences.

According to the properties of continuity and relativity of spatial substrates, some features of the unknown views can be inferred:

Hole Continuity: all the holes observed in a view affect the opposite views of the object, that is, front-back, right-left, up-down. Taking into account this property, the following features could be inferred regarding the *back*, *left* and *down* views of object 1:

Back	Left	Down
[*, -, -]	[-, *, *]	[-, -, -]
[-, -, -]	[-, -, -]	[-, -, -]
[-, -, -]	[-, -, -]	[-, -, -]

Note that features that remain unknown are represented by ‘-’.

Depth Relativity: features indicating the last level of depth in a view (i.e., level *c* in a 3x3x3 description), involve the existence of the first level of depth in the opposite view (always level *a*). Taking into account this property, the following features could be inferred regarding the *back*, *left* and *down* views of object 1:

Back	Left	Down
[*, a, a]	[-, *, *]	[-, -, -]
[a, -, -]	[-, -, -]	[a, -, -]
[-, -, -]	[-, -, -]	[a, -, a]

The consistency properties mentioned in Section V must be fulfilled also by the unknown perspectives and they can be used to infer the depths in them. For example, the neighbouring perspectives of *back* are *right*, *up*, and *left*. If *Right* and *Up* perspectives are known (R, U) or given by a *Q3D*, some features regarding the perspective *Left* can be inferred. These inference inter-relationships between neighbouring perspectives to discover more features in unknown views are currently under study.

VIII. IMPLEMENTATION

First-order logic knowledge bases are usually built using Horn clauses [32], which contains at most one positive literal. Prolog programming language [33] is based on Horn clause logic and it was selected as the logic programming language for implementing the logics of the *Q3D* description. SwiProlog³ was the testing platform [34], and the Prolog Contest book [35] was a guide.

The *Q3D* description of objects was written using Prolog facts as: `view(View, Object, N, Q3D)`.

For example, the *Q3D* description of the object in Fig. 3 is described as:

```
view(front,obj1,3,[[c,c,*],[*,*,b],[a,a,b]]).
view(right,obj1,3,[[b,a,c],[b,b,a],[b,b,c]]).
view(up,obj1,3,[[a,a,a],[a,a,a],[c,b,c]]).
```

The correctness of the input *Q3D* was checked. The consistency logics were also programmed and tested. The maximal volume of the objects regarding the *Q3D* was also programmed and tested. And the inference of some features of the unknown perspectives from their opposite perspectives were also programmed and checked.

As an example, the results of the Prolog implementation for the *Q3D* description of the object in Fig. 3 are given:

```
?- qualitative_3D(object1).
Front:[[c,c,*],[b,a,c],[a,a,a]] Correct description.
Right:[[*,*,b],[b,b,a],[a,a,a]] Correct description.
Up:[[a,a,b],[b,b,c],[c,b,c]] Correct description.
Consistent Q3D F,R,U views.
Maximal volume:17
Back constrained wrt Front:[[*,a,a],[a,?,?],[?,?,?]]
Left constrained wrt Right:[[?,*,*],[?,?,?],[?,?,?]]
Down constrained wrt Up:[[?,?,?],[a,?,?],[a,?,a]]
true.
```

More examples of the testings are provided in the Appendix. All the Prolog code corresponding to the *Q3D* is available for downloading⁴. For easily testing, the on-line platform Pengines⁵ can be used.

IX. DISCUSSION ABOUT RELATED WORK

In the literature, objects are also described using 3D shape grammars [36]. As linguistic grammars build sentences and paragraphs, shape grammars follow also rules (i.e., recursively subdivision) to build 3D objects. In these grammars, although the rules applied are logical, the obtained description of the object is not qualitative, in contrast to the one proposed in this paper.

Moreover, another approach related to shape grammars is constructive solid geometry [37] (or computational binary solid geometry), that is, a technique used in solid modelling which can define the steps of building/synthesising complex objects by combination of other objects and Boolean operations (i.e., intersection, union, difference). It has a broad application in computer graphics for generating objects in computer games [38].

Both shape grammars and constructive solid geometry methods are useful for object building/synthesis, but challenging to use for object description/analysis because they sometimes use not-reversible operations. Moreover, there is not a specific set of grammar rules or constructive geometry methods to obtain a specific object, since different methods can produce the same result. Therefore, the descriptions obtained might be not unique and then difficult to use for object identification. The *Q3D* model defined here can be useful for designing the plan to

⁴Data download: <https://sites.google.com/site/zfalomir/projects/cognitive.2ami>

⁵Pengines by SWI-Prolog: <http://pengines.swi.2prolog.org/apps/swish/index.html>

³SWI-Prolog: <http://www.swi.2prolog.org/>

synthesise/build the object, but also to uniquely describe that object when it is created.

In the literature, the main theoretical approaches in qualitative 3D representation which are studied in psychology of object perception are:

- Marr and Nishihara's approach [39] which uses a 3D composition of generalised cylinders to describe a sketch or skeleton of the objects and their parts. This model is hierarchical, that is, component parts can also be decomposed into parts and recognition is achieved when matching a description derived from an image to a previous stored 3D object type. The *Q3D* approach presented in this paper is similar to Marr and Nishihara's approach [39] in the sense that it uses a generalised cube to describe the structure of the object, similarly to Marr and Nishihara's cylinder. However the *Q3D* approach represented the whole object, not only its skeleton.
- Biederman's approach [40] describes 3D objects using more geometric shapes or geons, not only cylinders or cubes. However, they are obtained from a 2D image representation rather than from a 3D representation as in Marr and Nishihara's approach [39]. According to Biederman, geons are detected on the basis of certain properties of contours in the image (i.e., linearity, parallelism, curvilinearity, symmetry) or at regions of concavity. Therefore, Biederman's approach tries also to represent the different shapes of the components of the object, not only the volume parts.
- Guesgen's approach [1] approximate objects to polygons with parallel sides which are projected to a coordinate axis. The relations between the objects (or intervals in the axis) are represented qualitatively (i.e., left of, attached to, overlapping, inside) similarly to the relations between temporal intervals defined by Allen's model [41]. This model is similar to the *Q3D* in the sense that both approximate objects, Guesgen's approach to polygons and the *Q3D* to arrangements of cubes. However, the *Q3D* uses the cube as a unit which allows to calculate the volume of the object, whereas Guesgen's method do not obtain it, but it is independent of it.

The approaches above describe objects based on their 3D structural skeletons or sides and produce object centred descriptions. That is, view-independent descriptions are obtained which are not designed to detect inconsistencies in object perception from different views. The *Q3D* approach presented here describes the depth of 3D objects in its canonical views. The representation obtained is object-centred, but allows comparison between perspectives in order to detect inconsistencies and also in order to infer unknown perspectives, which is a novel aspect in the literature, as far as we are concerned. Moreover, there is support for the general idea that human object recognition involves view-dependent representations, that is, people prefer to imagine, view, or photograph objects from certain "canonical" views [30]. Therefore, the *Q3D* has a cognitive basis.

X. CONCLUSION AND FUTURE WORK

This paper presents the definition of a qualitative 3D object descriptor based on a qualitative concept of depth which considers a cube as the minimal unit of volume. This representation

can be considered a spatial substrate [31], since if a single cube is added or removed at an edge, this change is produced for free to 3 perspectives (i.e., *front*, *right* and *up*) without no computing effort needed to readjust the views.

Abstraction, continuity and interrelation properties are proposed in this paper as the basis to define spatial substrates. The logics to test the consistency of the Front, Right and Up *Q3D* descriptions corresponding to real 3D objects are presented. Then, the maximal volume of an object is calculated logically from the *Q3D* obtained. Moreover, logics to infer some features of the unknown back, down and left views are proposed.

All the *Q3D* logics described above have been implemented in Prolog and tested using the SwiProlog platform. Results are promising and they inspire future work towards an approach which could give humans a hint about which projections of a 3D object are impossible when solving spatial problems, so that they could understand and reason about 3D object representations such as those in the test by the German Academic Foundation for scholarships².

As future work, it is intended to: (i) implement further reasoning methods to infer the rest of features of depth of the unknown *back*, *left* and *down* perspectives, from the data known regarding its neighbouring perspectives; (ii) extend the *Q3D* description to include hidden concavities; and (iii) define an approach to describe the boundary shape of each of the perspectives of the object taking into account the *Q3D* as a basis.

Applications in education are envisioned when helping students in engineering to understand conventions in technical drawing. Other applications in computer vision would be interesting, for example when computing 3D attention saliency in proto-objects [42], a *Q3D* description could help to store a short memory narrative of the evolution of the proto-objects in these attention artificial systems.

ACKNOWLEDGMENTS

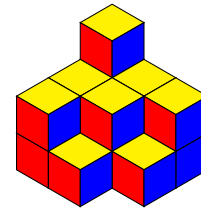
Dr.-Ing. Zoe Falomir acknowledges funding by the project COGNITIVE-AMI (GA 328763) by the European Commission through FP7 Marie Curie IEF actions and the support by the Universität Bremen and the Spatial Cognition Centre.

The author also thanks the anonymous reviewers' comments which helped to improve this paper and the plane partition latex style fonts by Jang Soo Kim.

APPENDIX

Other examples of *Q3Ds* implemented in Prolog and used in the testings.

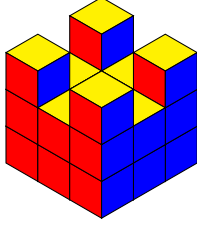
Object 2:



Front	Right	Up	Back	Left	Down
$[c, *, *]$	$[*, *, c]$	$[a, b, b]$	$[*, *, a]$	$[a, *, *]$	$[a, a, a]$
$[a, b, c]$	$[c, b, a]$	$[b, b, c]$	$[a, a, a]$	$[a, a, a]$	$[a, a, a]$
$[a, a, b]$	$[b, a, a]$	$[b, c, *]$	$[a, a, a]$	$[a, a, a]$	$[*, a, a]$

```
?- qualitative_3D(object2).
Front: [[c,*,*],[a,b,c],[a,a,b]] Correct description.
Right: [[*,*,c],[c,b,a],[b,a,a]] Correct description.
Up: [[a,b,b],[b,b,c],[b,c,*]] Correct description.
Consistent Q3D F,R,U views.
Maximal volume:15
true .
```

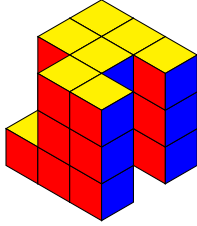
Object 3:



Front	Right	Up	Back	Left	Down
[a,*,a]	[a,*,a]	[a,b,a]	[a,*,a]	[a,*,a]	[a,a,a]
[a,a,a]	[a,a,a]	[b,b,b]	[a,a,a]	[a,a,a]	[a,a,a]
[a,a,a]	[a,a,a]	[a,b,a]	[a,a,a]	[a,a,a]	[a,a,a]

```
?- qualitative_3D(object3).
Front: [[a,*,a],[a,a,a],[a,a,a]] Correct description.
Right: [[a,*,a],[a,a,a],[a,a,a]] Correct description.
Up: [[a,b,a],[b,b,b],[a,b,a]] Correct description.
Consistent Q3D F,R,U views.
Maximal volume:22
true .
```

Object 4:



Front	Right	Up	Back	Left	Down
[b,a,a]	[a,b,a]	[a,a,a]	[a,a,a]	[a,a,b]	[a,a,a]
[b,a,a]	[a,b,a]	[a,a,*]	[a,a,a]	[a,a,b]	[*,a,a]
[a,a,a]	[a,b,a]	[c,a,a]	[a,a,a]	[a,a,a]	[a,a,a]

```
?- qualitative_3D(object4).
Front: [[b,a,a],[b,a,a],[a,a,a]] Correct description.
Right: [[a,b,a],[a,b,a],[a,b,a]] Correct description.
Up: [[a,a,a],[a,a,*],[c,a,a]] Correct description.
Consistent Q3D F,R,U views.
Maximal volume:22
true .
```

REFERENCES

- [1] H. W. Guesgen, "Spatial Reasoning Based on Allen's Temporal Logic," International Computer Science Institute, Tech. Rep., 1989.
- [2] A. G. Cohn and J. Renz, *Qualitative Spatial Reasoning, Handbook of Knowledge Representation*, V. L. F. Harmelen and B. Porter, Eds. Wiley-ISTE, London: Elsevier, 2007.
- [3] G. Ligozat, *Qualitative Spatial and Temporal Reasoning*. Wiley-ISTE, London: MIT Press, 2011.
- [4] L. Kunze, C. Burbridge, and N. Hawes, "Bootstrapping probabilistic models of qualitative spatial relations for active visual object search," in *Qualitative Representations for Robots, Proc. AAAI Spring Symposium, Technical Report SS-14-06*, 2014, pp. 81–80, ISBN 978-1-57735-646-2.
- [5] Z. Falomir, L. Museros, V. Castelló, and L. Gonzalez-Abril, "Qualitative distances and qualitative image descriptions for representing indoor scenes in robotics," *Pattern Recognition Letters*, vol. 38, pp. 731–743, 2013. [Online]. Available: <http://dx.doi.org/10.1016/j.patrec.2012.08.012>
- [6] Z. Falomir, E. Jiménez-Ruiz, M. T. Escrig, and L. Museros, "Describing images using qualitative models and description logics," *Spatial Cognition and Computation*, vol. 11, no. 1, pp. 45–74, 2011. [Online]. Available: <http://dx.doi.org/10.1080/13875868.2010.545611>
- [7] A. Cohn, D. Hogg, B. Bennett, V. Devin, A. Galata, D. Magee, C. Needham, and P. Santos, "Cognitive vision: Integrating symbolic qualitative representations with computer vision," in *Cognitive Vision Systems*, ser. Lecture Notes in Computer Science, H. Christensen and H.-H. Nagel, Eds. Springer Berlin/Heidelberg, 2006, vol. 3948, pp. 221–246.
- [8] M. Bhatt and F. Dylla, "A qualitative model of dynamic scene analysis and interpretation in ambient intelligence systems," *I. J. Robotics and Automation*, vol. 24, no. 3, 2009. [Online]. Available: <http://dx.doi.org/10.2316/Journal.206.2009.3.206.23274>
- [9] Z. Falomir and A.-M. Olteanu, "Logics based on qualitative descriptors for scene understanding," *Neurocomputing*, vol. 161, pp. 3–16, 2015. [Online]. Available: <http://dx.doi.org/10.1016/j.neucom.2015.01.074>
- [10] Z. Falomir, L. Gonzalez-Abril, L. Museros, and J. Ortega, "Measures of similarity between objects from a qualitative shape description," *Spatial Cognition and Computation*, vol. 13, pp. 181–218, 2013. [Online]. Available: <http://dx.doi.org/10.1080/13875868.2012.700463>
- [11] Z. Falomir, L. Museros, and L. Gonzalez-Abril, "A model for colour naming and comparing based on conceptual neighbourhood. An application for comparing art compositions," *Knowledge-Based Systems*, vol. 81, pp. 1–21, 2015. [Online]. Available: <http://dx.doi.org/10.1016/j.knsys.2014.12.013>
- [12] M. Bhatt and C. Freksa, "Spatial computing for design an artificial intelligence perspective," in *Studying Visual and Spatial Reasoning for Design Creativity*, J. S. Gero, Ed., 2015, pp. 109–127.
- [13] P. Fogliaroni, *Qualitative Spatial Configuration Queries. Towards Next Generation Access Methods for GIS*, ser. Dissertations in Geographic Information Science. IOS Press, 2013, ISBN 978-1614992486.
- [14] R. Al-Salman, "Qualitative spatial query processing: Towards cognitive geographic information systems," Ph.D. dissertation, University of Bremen, 2014, supervised by Prof. Christian Freksa (University of Bremen) and Prof. Christian Jensen (Aalborg University).
- [15] A. Lovett, M. Dehghani, and K. Forbus, "Learning of qualitative descriptions for sketch recognition," in *Proc. 20th Int. Workshop on Qualitative Reasoning (QR)*, Hanover, USA, July, 2006.
- [16] A. Lovett and K. Forbus, "Cultural commonalities and differences in spatial problem-solving: A computational analysis," *Cognition*, vol. 121, no. 2, pp. 281 – 287, 2011. [Online]. Available: <http://dx.doi.org/10.1016/j.cognition.2011.06.012>
- [17] I. Kazmi, L. You, and J. J. Zhang, "A Survey of 2D and 3D Shape Descriptors," in *Computer Graphics, Imaging and Visualization (CGIV), 2013 10th International Conference*, Aug 2013, pp. 1–10. [Online]. Available: <http://dx.doi.org/10.1109/CGIV.2013.11>
- [18] J. Tangelder and R. Veltkamp, "A survey of content based 3d shape retrieval methods," *Multimedia Tools and Applications*, vol. 39, no. 3, pp. 441–471, 2008. [Online]. Available: <http://dx.doi.org/10.1007/s11042.2007.20181.20>
- [19] T. Kluth and Z. Falomir, "Studying the role of location in 3D scene description using natural language," in *XV Workshop of the Association on Qualitative Reasoning and its Applications (JARCA13). Qualitative Systems and their applications to Diagnosis, Robotics and Ambient Intelligence*, I. Sanz, L. Museros, and J. A. Ortega, Eds. Murcia, Spain: Proceedings from the University of Seville, 2013, pp. 33–36.
- [20] K. Lai, L. Bo, X. Ren, and D. Fox, "Sparse Distance Learning for Object Recognition Combining RGB and Depth Information," in *IEEE International Conference on on Robotics and Automation*, 2011. [Online]. Available: <http://dx.doi.org/10.1109/ICRA.2011.5980377>
- [21] L. Bo, X. Ren, and D. Fox, "Depth kernel descriptors for object recognition," in *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2011, San Francisco, CA, USA, September 25-30*. IEEE, 2011, pp. 821–826. [Online]. Available: <http://dx.doi.org/10.1109/IROS.2011.6095119>
- [22] N. Newcombe, "Picture this: Increasing math and science learning by improving spatial thinking," *American Educator*, vol. 34, no. 2, pp. 29–35, 2010.
- [23] J. Wai, D. Lubinski, and C. P. Benbow, "Spatial ability for STEM domains: Aligning over 50 years of cumulative psychological knowledge solidifies its importance," *Journal of Educational Psychology*, vol. 101, no. 4, pp. 817–835, 2009. [Online]. Available: <http://dx.doi.org/10.1037/a0016127>
- [24] N. Sinclair, E. de Freitas, and F. Ferrara, "Virtual encounters: the murky and furtive world of mathematical inventiveness," *ZDM. The International Journal on Mathematics Education*, vol. 45, no. 2, pp. 239–252, 2013. [Online]. Available: <http://dx.doi.org/10.1007/s11858.2012.20465.23>
- [25] N. Sinclair and C. D. Bruce, "Spatial reasoning for young learners, Research Forum," in *Proc. of the 38th Conference of the International Group for the Psychology of Mathematics Education and the 36th Conference of the North American Chapter of the Psychology of Mathematics Education (PME 38 / PME-NA 36)*, Vancouver, Canada, July 2014, pp. 173–205.

- [26] K. Highfield and J. Mulligan, "The role of dynamic interactive technological tools in preschoolers' mathematical patterning," in *Proc. of the 30th annual conference of the Mathematics Education Research Group of Australasia*, J. Watson and K. Beswick, Eds., vol. 1. MERGA, 2007, pp. 372–381. [Online]. Available: <http://hdl.handle.net/1959.14/150138>
- [27] L. Museros, Z. Falomir, I. Sanz, and L. Gonzalez-Abril, "Sketch retrieval based on qualitative shape similarity matching: Towards a tool for teaching geometry to children," *AI Communications*, vol. 28, no. 1, pp. 73–86, 2014. [Online]. Available: <http://dx.doi.org/10.3233/AIC.2140614>
- [28] M. A. Menchaca-Brandan, A. M. Liu, C. M. Oman, and A. Natapoff, "Influence of perspective-taking and mental rotation abilities in space teleoperation," in *Proceedings of the ACM/IEEE International Conference on Human-robot Interaction*, ser. HRI '07. New York, NY, USA: ACM, 2007, pp. 271–278. [Online]. Available: <http://doi.acm.org/10.1145/1228716.1228753>
- [29] M. Hegarty, M. Keehner, C. A. Cohen, D. R. Montello, and Y. Lippa, "The role of spatial cognition in medicine: Applications for selecting and training professionals," in *Applied spatial cognition: From research to cognitive technology*, G. L. Allen, Ed. Mahwah, NJ: Lawrence Erlbaum, 2007, pp. 285–315.
- [30] S. Palmer, E. Rosch, and P. Chase, "Canonical perspective and the perception of objects," *Attention and Performance IX*, pp. 135–151, 1981.
- [31] C. Freksa, "Spatial computing – how spatial structures replace computational effort," in *Cognitive and linguistic aspects of geographic space*, M. Raubal, D. Mark, and A. F. (Eds.), Eds. Heidelberg: Springer, 2013.
- [32] J. W. Lloyd, *Foundations of logic programming. Symbolic computation: Artificial intelligence*. Springer-Verlag, 2nd, extended edition, 1987.
- [33] L. Sterling and E. Shapiro, *The Art of Prolog (2nd Ed.): Advanced Programming Techniques*. Cambridge, MA, USA: MIT Press, 1994.
- [34] J. Wielemaker, T. Schrijvers, M. Triska, and T. Lager, "SWI-Prolog," *Theory and Practice of Logic Programming (TPLP)*, vol. 12, no. 1-2, pp. 67–96, 2012. [Online]. Available: <http://dx.doi.org/10.1017/S1471068411000494>
- [35] B. Demoen, P.-L. Nguyen, T. Schrijvers, and R. Tronçon, *The First Programming Contests*, Belgium, 2005. [Online]. Available: <http://www.cs.kuleuven.be/~dtai/ppcbook/>
- [36] A. I.-K. Li, L. Chen, Y. Wang, and H. Chau, "Editing Shapes in a Prototype Two-and Three-dimensional Shape Grammar Environment," in *Computation: The New Realm of Architectural Design: 27th eCAADe Conference Proceedings*, ser. eCAADe: Conferences. Istanbul, Turkey: Istanbul Technical University, Faculty of Architecture, 2009, pp. 243–250.
- [37] A. A. G. Requicha, "Representations for rigid solids: Theory, methods, and systems," *Computing Surveys*, vol. 12, no. 4, pp. 437–464, Dec. 1980.
- [38] S. van Rossen and M. Baranowski, *Real-Time Constructive Solid Geometry*. A K Peters/CRC Press, 2011, pp. 79–96. [Online]. Available: <http://dx.doi.org/10.1201/b10946.211>
- [39] D. Marr and H. K. Nishihara, "Representation and Recognition of the Spatial Organization of Three-Dimensional Shapes," *Proceedings of the Royal Society of London. Series B. Biological Sciences*, vol. 200, no. 1140, pp. 269–294, 1978.
- [40] I. Biederman, "Recognition-by-components: A theory of human image understanding," *Psychological Review*, vol. 94, pp. 115–147, 1987.
- [41] J. F. Allen and L. F. Allen, "Maintaining knowledge about temporal intervals," *Communication of ACM*, pp. 832–843, 1983.
- [42] P. Lanillos, J. F. Ferreira, and J. Dias, "Multisensory 3D saliency for artificial attention systems," in *Proc. of the 3rd Workshop on Recognition and Action for Scene Understanding (REACTS)*, Malta, G. Azzopardi, F. Escolano, and R. Marfil, Eds., 2015, pp. 1–14, ISBN 978-84-606-9592-9.

Spatial Rules for Capturing Qualitatively Equivalent Configurations in Sketch maps

Sahib Jan, Carl Schultz, Angela Schwering and Malumbo Chipofya

Institute for Geoinformatics

University of Münster, Germany

Email: sahib.jan | schultzc | schwering | mchipofyal@uni-muenster.de

Abstract—Sketch maps are an externalization of an individual's mental images of an environment. The information represented in sketch maps is schematized, distorted, generalized, and thus processing spatial information in sketch maps requires plausible representations based on human cognition. Typically only qualitative relations between spatial objects are preserved in sketch maps, and therefore processing spatial information on a qualitative level has been suggested. This study extends our previous work on qualitative representations and alignment of sketch maps. In this study, we define a set of spatial relations using the declarative spatial reasoning system CLP(QS) as an approach to formalizing key spatial aspects that are preserved in sketch maps. Unlike geo-referenced maps, sketch maps do not have a single, global reference frame. Rather, the sketched elements themselves act as referencing objects. Using the declarative spatial reasoning system CLP(QS), we define constraint logic programming rules that formalize various key spatial aspects of sketch maps at a local level, between nearby objects. These rules focus on linear ordering, cyclic ordering, and relative orientation of depicted objects along, and around, salient reference objects.

I. INTRODUCTION

SKETCH maps are used to externalize an individual's mental image of the environment. The information represented in sketch maps is based on observation rather than measurements. Therefore, information in sketch maps is schematized, distorted, and generalized. Freehand sketch maps contain objects and spatial relations between these objects which enable users to use sketch maps to communicate and reason about actions in environments. During the last two decades, several approaches [7, 9, 21, 32] attempt to capture spatial configurations between depicted objects qualitatively. These approaches use different aspects of space such as topological relations, cardinal directions, relative orientations, and relative distances.

Throughout a series of experiments [27, 33, 34], Wang et al. identify a set of sketch aspects which are not subject to schematizations, distortions or any other cognitive impact. These sketch aspects represent: linear ordering, cyclic ordering, relative orientations, and topological relations between spatial objects. The identified sketch aspects are categorized into local and global levels [27]. The local level relations refer to the relations between nearby objects while global level relations represent possible relations between all objects in a map. During the last two decades, a series of qualitative spatial calculi have been proposed in the area of Qualitative Spatial Reasoning (QSR) [11] to formalize some of these aspects

such as representations for the topological relations [6, 23], orderings [1, 22, 25], directions [10, 24], relative position of points [19, 20, 24] and others.

In our previous studies [14, 15, 16, 27], we propose a set of plausible representations and their coarse versions to qualitatively formalize key sketch aspects. We use several *qualifiers* to extract qualitative constraints from geometric representations of sketch and geo-referenced maps [13] in the form of Qualitative Constraint Networks (QCNs). QCNs are complete graphs representing spatial objects and relations between them. However, in order to derive more cognitively accurate QCNs, we require greater flexibility in being able to define qualitative spatial relations for our particular application domain, i.e. geographic-scale sketch maps. Specifically, sketch maps require qualitative constraints at a local level between particular types of adjacent objects such as linear ordering, cyclic ordering, and orientation information of nearby landmarks with respect to reference objects.

In this study, we propose the utilisation of the declarative spatial reasoning system CLP(QS) [2, 26] as an alternative approach to deriving cognitively plausible constraints between nearby objects. The system is capable of modeling and reasoning about qualitative spatial relations within the context of the constraint logic programming. Using the CLP(QS) framework, we define logic programming rules over qualitative spatial domains in order to express and solve declarative, high-level constraints between spatial objects depicted in sketch maps.

The remainder of this paper is structured as follows: In the following section, we briefly introduce related work. In Section 3 we discuss spatial objects and cognitively plausible aspects found in sketch maps. In Section 4 we present CLP(QS) rules that formalize the cognitively salient aspects of sketch maps. Section 5 concludes the paper with an outlook on future work.

II. RELATED WORK

The information in sketch maps is based on observations rather than measurements. Therefore, processing spatial information on a qualitative level has been suggested [5, 27]. During the last two decades, several approaches [7, 9, 21, 32] attempt to capture spatial configurations between depicted objects qualitatively. Egenhofer et al. [8] propose Spatial-Query-by-Sketch, a sketch-based user interface that focuses on enabling a user to specify spatial relations (topology and

cardinal directions) by drawing them. Volker et al. [31] propose the visual query system VISCO. It offers a sketch-based query language for defining approximate spatial constellations of the objects. Forbus et al. [9] develop a sketch understanding system CogSketch which is a space search system that focuses on topological relations among sketched elements, and reasons about these relations to infer new knowledge. Nedas et al. [21] propose a *similarity* measure methodology for comparing two spatial scenes by identifying cognitively-motivated similarities between objects, relations among spatial objects, and the ratio of the total number of objects in both scenes to the number of objects that have been correctly matched.

These approaches share our motivation of using abstract qualitative relations to represent spatial configurations between objects depicted by a user. In a previous study [27], we propose a framework to preprocess, align and integrate sketched spatial information on a qualitative level. The framework addresses the extraction of objects from sketch maps, computing QCNs from geometric representations of sketch and geo-referenced maps, and aligning them qualitatively.

This study extends our previous work on qualitative representations of spatial objects. As the outline of spatial objects in freehand sketches are imprecise, the qualitative representation of spatial objects with imprecise boundaries leads to different qualitative relations when compared to relations in geo-referenced maps. In this study, we present spatial rules defined using the CLP(QS) system as an alternative approach to compute qualitative relations (on a conceptual level) which are preserved in freehand sketch maps. These rules address linear ordering, cyclic ordering, and orientation information of adjacent objects along and around key reference objects.

III. SPATIAL OBJECTS AND THEIR CONFIGURATIONS IN SKETCH MAPS

A. Spatial objects

Inspired by the Lynch's seminal work [18] and Tversky's analysis of mental structures [30], we characterize the depicted objects in sketch maps into four elements: streets segments, junctions, landmarks, and city-blocks. These elements are automatically extracted using the object recognition method proposed in [4].

Street segments are connected, and mostly linear, features in sketch maps. They are represented as line segments and are connected to other street segments at junctions. The connectivity of street segments and the street-network is central for human path planning [12]. Junctions are the end-points of street segments. The end-points, where street segments are not connected to other street segments, are called hanging end-points. At the boundary of the sketching medium, street segments are left unconnected to any further street segment, resulting in hanging end-points. In sketch maps, junctions capture the connectivity of various street segments forming a street network. In our approach, both hanging end-points and junctions are spatially represented as 2D points.

Landmarks are the most salient elements in an environment and are therefore essential to characterize an environment. In

freehand sketches, landmarks are vectorized and approximated by polygons which represent spatial entities such as water bodies, buildings, and parks. Landmarks and road entities are the most frequently depicted spatial objects in sketch maps [3], while city-blocks are the smallest regions. They are delimited by a lineal representation of connected street segments. People do not always sketch complete city-blocks, in particular at the edge of the sketch medium. Therefore, we define city-blocks as areas either bounded by the street segments, or bounded by street segments and the boundary of the medium [14].

B. Invariant spatial aspects in Sketch Maps

Processing sketch information at a qualitative level requires explicit knowledge about certain aspects of sketches that are not subject to schematizations, distortions, or any other cognitive impact [29]. That is, these aspects are preserved in freehand sketches. Throughout a series of experiments [27, 33, 34], Wang et al. identified a set of seven invariant sketch aspects. These aspects consist of: linear ordering of landmarks and street segments along a route, cyclic ordering of landmarks and street segments around reference junctions, relative orientation of landmarks with respect to street segments, topological and orientation relations between street segments in street network, and topological relations between extended objects (landmarks, and city-blocks). This paper focuses on the formalization of ordering and relative orientation of adjacent objects using spatial rules defined in the context of the CLP(QS) system.

IV. SPATIAL RULES FOR QUALITATIVELY EQUIVALENT CONFIGURATIONS

Using the CLP(QS) framework, we define spatial rules to compute qualitative information between nearby objects. For the linear ordering and orientation information of adjacent landmarks, we use connected street segments as reference objects, while junctions are used as reference objects for cyclic ordering. The adjacency of landmarks is defined via relative metric distances.

A. Preliminaries

CLP(QS) includes a library of qualitative spatial relations encoded as polynomial constraints over a set of real variables X , which are solved via constraint logic programming [2]. In this subsection we present the CLP(QS) library implementations of projection, distance, and orientation relations that we build on in subsequent sections. A set of spatial relations is *consistent* in CLP(QS) if there exists some assignment of reals to the variables X such that all of the corresponding polynomial constraints are satisfied. CLP(QS) uses a variety of polynomial solvers including CLP(R), SAT Modulo Theories, quantifier elimination by Cylindrical Algebraic Decomposition, and geometric constraint solvers.

Projection. A point is projected onto a line using the dot product. This is extended to segment-line projection by projecting both end points. Polygons are projected onto lines by projecting all vertices and taking the maximum and minimum

projected values as the projected interval. Points are projected onto *segments* by clamping the projected value to lie within the projection of the segment and the line collinear with the segment, i.e. let v, a, b be reals such that $a \leq b$ then

$$\text{CLAMP}(v, a, b) = \begin{cases} v & \text{if } a \leq v \leq b \\ a & \text{if } v < a \\ b & \text{if } v > b \end{cases}$$

CLP(QS) projection predicates are implemented as follows.

```
projection(value(V), point(Xp,Yp),
           line(point(Xa,Ya),point(Xb,Yb))) :-
  {(Xp - Xa) * (Xb - Xa) + (Yp - Ya) * (Yb - Ya) == V}.
```

```
projection(interval(Prj1,Prj2),
           segment(Pa1,Pa2), segment(Pb1,Pb2)) :-
  projection(value(Prj1), Pa1, segment(Pb1,Pb2)),
  projection(value(Prj2), Pa2, segment(Pb1,Pb2)).
```

```
projection(value(V), point(Xp,Yp),
           segment(point(Xa,Ya),point(Xb,Yb))) :-
  {(Xp - Xa) * (Xb - Xa) + (Yp - Ya) * (Yb - Ya) == PrjV},
  {Max := (Xb - Xa)^2 + (Yb - Ya)^2},
  clpqs_utils:clamp(0, Max, PrjV, V).
```

The CLP(QS) predicate for obtaining the coordinates of the projection onto a segment (or line) is:

```
reconstruct_projection(point(Xp,Yp), value(V),
                      segment(point(Xa,Ya),point(Xb,Yb))) :-
  {Limit := (Xb - Xa)^2 + (Yb - Ya)^2},
  {Limit > 0},
  {NormV := V / Limit},
  {Xp := (Xb - Xa) * NormV + Xa},
  {Yp := (Yb - Ya) * NormV + Ya}.
```

Euclidean distance. We employ CLP(QS) Euclidean distances between points, and between points and segments.

```
distance(value(V), point(Xa,Ya), point(Xb,Yb)) :-
  {(Xa - Xb)^2 + (Ya - Yb)^2 == V^2, V > 0}.
```

```
distance(nearer_than,
         point(Xa,Ya), point(Xb,Yb), point(Xr,Yr)) :-
  {(Xa - Xr)^2 + (Ya - Yr)^2 < (Xb - Xr)^2 + (Yb - Yr)^2}.
```

```
distance(equidistant,
         point(Xa,Ya), point(Xb,Yb), point(Xr,Yr)) :-
  {(Xa - Xr)^2 + (Ya - Yr)^2 == (Xb - Xr)^2 + (Yb - Yr)^2}.
```

```
distance(farther_than,
         point(Xa,Ya), point(Xb,Yb), point(Xr,Yr)) :-
  {(Xa - Xr)^2 + (Ya - Yr)^2 > (Xb - Xr)^2 + (Yb - Yr)^2}.
```

```
distance(value(V), point(Xp,Yp),
         segment(point(Xa,Ya), point(Xb,Yb))) :-
  projection(point(Xpj,Ypj), point(Xp,Yp),
            segment(point(Xa,Ya), point(Xb,Yb))),
  distance(value(V), point(Xpj,Ypj), point(Xp,Yp)).
```

Relative Orientation. We employ CLP(QS) relative orientation predicates between points and lines.

```
orientation(left_of, point(Xp,Yp),
            line(point(Xa,Ya), point(Xb,Yb))) :-
  {(Xb - Xa) * (Yp - Ya) > (Yb - Ya) * (Xp - Xa)}.
```

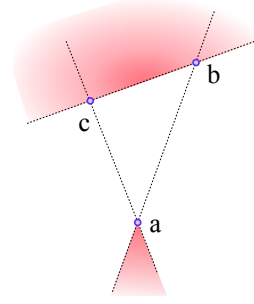


Fig. 1. Inconsistent relative orientation relations between triangle edges A, B, C and a point D : the point D must occupy the intersection of the red regions. As the red regions are disconnected, the constraints are unsatisfiable.

```
orientation(collinear, point(Xp,Yp),
            line(point(Xa,Ya), point(Xb,Yb))) :-
  {(Xb - Xa) * (Yp - Ya) == (Yb - Ya) * (Xp - Xa)}.
```

```
orientation(right_of, point(Xp,Yp),
            line(point(Xa,Ya), point(Xb,Yb))) :-
  {(Xb - Xa) * (Yp - Ya) < (Yb - Ya) * (Xp - Xa)}.
```

Spatial reasoning with incomplete numerical information.

We emphasise that, using CLP(QS), it is possible to reason about sets of spatial relations in the partial or complete absence of numerical information. For example, we can define an anti-clockwise triangle with vertices A, B, C with the constraint that C is on the *left of* line (A, B) . A point D can then be constrained by relative orientation relations with the triangle edges:

```
?- orientation(left_of, C, line(A,B)),
   | orientation(left_of, D, line(A,C)),
   | orientation(right_of, D, line(A,B)),
   | orientation(right_of, D, line(B,C)).
false.
...
| orientation(left_of, D, line(B,C)).
true.
```

Even though we have not provided any numerical information about the positions of the points A, B, C, D , CLP(QS) correctly determines that D cannot be simultaneously *left of* (A, C) and *right of* (A, B) and (B, C) as illustrated in Figure 1. We use this feature to reason about possible scenarios in sketch maps in cases where only incomplete numerical information is available.

B. Rules for Linear Ordering as Constraints

In sketch maps, linear ordering of spatial objects is an invariant sketch aspect [33]. It describes the linear ordering of adjacent landmarks and street segments along a route. A route is defined as a set of connected street segments. In [33, 34] Wang et al. found the linear ordering of both landmarks and street segments as a suitable representation for sketch map alignment. In our previous studies [16], we proposed a coarse version of Allen's interval algebra [1] to extract the linear ordering of spatial objects along a route.

We define spatial rules to compute the linear ordering of adjacent landmarks and connected street segments along a

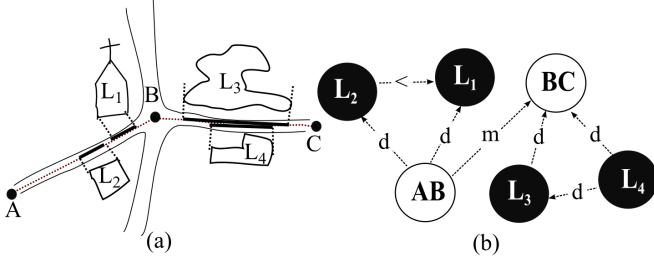


Fig. 2. (a) Spatial objects in the sketch map (b) Ordering of adjacent landmarks and connecting street segments along the route (AB,BC).

route. The adjacency of landmarks is computed using the relative metric distance between street segments and landmarks. A landmark is considered adjacent or local if its footprint intersects with the buffer around the reference street segment. Next, adjacent landmarks are projected from both sides of a route onto the street segments. As proposed in our previous studies [16], the ordering between intervals of projected landmarks and street segments are represented using both Allen's Interval relations and a coarse version of Allen's interval relations [1]. This is accomplished by encoding qualitative spatial relations as rules and facts using CLP(QS), as illustrated in Figure 2. Figure 2b shows the linear ordering of the depicted objects in terms of Allen's relations such as meets(m), before(<), and during(d).

```
ordering(Rel, St, Lm) :-
  object(St, type(street)),
  object(Lm, type(landmark)),
  relation(proximity(adjacent, Lm, St)),
  oriented_representation(St, segment(Pa, Pb)),
  representation(Lm, GLm),
  projection(PrjLm, GLm, line(Pa, Pb)),
  projection(PrjSt, segment(Pa, Pb), line(Pa, Pb)),
  allen(Rel, PrjSt, PrjLm).
```

Buffer Size. In sketch maps, the distances between depicted objects are distorted. The distances between landmarks and adjacent street segments are always variant between sketch maps of the same location. Therefore, defining a common buffer size for all freehand sketch maps is not possible. In order to compute relative buffer, we define an automatic method as rule in CLP(QS) which computes minimum distances between landmarks and street segments and then considers the maximum distance as buffer size from the computed minimum distances.

```
set_buffer_distance :-
  max_landmark_street_distance(value(D)),
  retractall(buffer(_)), assert(buffer(D)).

max_landmark_street_distance(value(D)) :-
  setof(Dist,
    L^nearest_street_distance(value(Dist), L, Dists),
    sort(Dists, SDists), reverse(SDists, [D|_])).

nearest_street_distance(value(MinD), L) :-
  object(L, type(landmark)), representation(L, LG),
  setof(D,
    S^SG^(object(S, type(street)), representation(S, SG),
    distance(value(D), LG, SG), Ds),
    sort(Ds, [MinD|_]), write(' '), flush.
```

Adjacency. We define a rule for adjacency as a qualitative

constraint between landmarks and street segments. A landmark is considered adjacent when the polygonal footprint of the landmark intersects with the buffer of the reference street segment. The adjacency rule is used to define linear ordering locally, between nearby objects.

```
adjacent(Lm, St) :-
  object(Lm, type(landmark)),
  object(St, type(street)), buffer(Buf),
  representation(St, StreetGeom),
  representation(Lm, LandmarkGeom),
  distance(value(Dist), LandmarkGeom, StreetGeom),
  Dist < Buf.
```

Linear Projection. This spatial rule projects landmarks onto adjacent street segments. The projections of landmarks onto street segments are represented as pairs of intervals with start and end points. The rule also projects spatial entities such as point-to-line, line-to-line, and polygon-to-line objects. The projected intervals together with the intervals of street segments provide ordering information between spatial objects along a route.

```
projection(point(Xpj, Ypj), point(Xp, Yp),
  segment(Pa, Pb)) :-
  projection(
    value(V), point(Xp, Yp), segment(Pa, Pb)),
  reconstruct_projection(
    point(Xpj, Ypj), value(V), segment(Pa, Pb)).
```

Qualitative Interval Ordering. These spatial rules define linear ordering relations between projected intervals. The rule compares the start and end points of two intervals and represents them as relations as defined in Allen's Interval Algebra (IA).

```
allen(before, interval(A1, A2), interval(B1, B2)) :-
  {A1 < A2, B1 < B2},
  {A2 < B1}.
```

```
allen(after, interval(A1, A2), interval(B1, B2)) :-
  allen(before, interval(B1, B2), interval(A1, A2)).
```

```
allen(meets, interval(A1, A2), interval(B1, B2)) :-
  {A1 < A2, B1 < B2},
  {A2 =:= B1}.
```

```
allen(met_by, interval(A1, A2), interval(B1, B2)) :-
  allen(meets, interval(B1, B2), interval(A1, A2)).
```

```
allen(overlaps, interval(A1, A2), interval(B1, B2)) :-
  {A1 < A2, B1 < B2},
  {A2 > B1, A1 < B1, A2 < B2}.
```

```
allen(overlapped_by, interval(A1, A2), interval(B1, B2)) :-
  allen(overlaps, interval(B1, B2), interval(A1, A2)).
```

```
allen(starts, interval(A1, A2), interval(B1, B2)) :-
  {A1 < A2, B1 < B2},
  {A1 =:= B1, A2 < B2}.
```

```
allen(started_by, interval(A1, A2), interval(B1, B2)) :-
  allen(starts, interval(B1, B2), interval(A1, A2)).
```

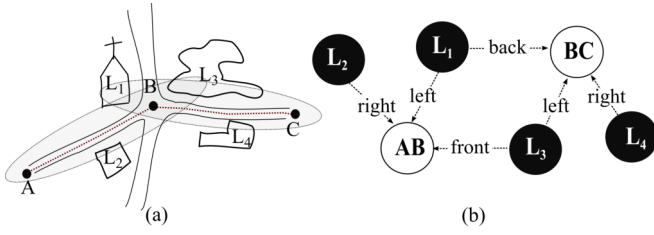


Fig. 3. (a) Orientation of adjacent landmarks with respect to street segments
(b) Qualitative constraints representing orientation relations.

```
allen(during, interval(A1,A2), interval(B1,B2)) :-
  {A1 < A2, B1 < B2},
  {A1 > B1, A2 < B2}.
```

```
allen(contains, interval(A1,A2), interval(B1,B2)) :-
  allen(during, interval(B1,B2), interval(A1,A2)).
```

```
allen(finishes, interval(A1,A2), interval(B1,B2)) :-
  {A1 < A2, B1 < B2},
  {A1 > B1, A2 =:= B2}.
```

```
allen(finished_by, interval(A1,A2), interval(B1,B2)) :-
  allen(finishes, interval(B1,B2), interval(A1,A2)).
```

```
allen(equal, interval(A1,A2), interval(B1,B2)) :-
  {A1 < A2, B1 < B2},
  {A1 =:= B1, A2 =:= B2}.
```

C. Rules for Relative Orientation of Landmarks

It is common to use points as basic entities in positional reasoning [11, 19]. In [14], we investigate different qualitative representations in order to formalize orientation information of landmarks with respect to adjacent street segments and relations defined in the Left-Right (LR) calculus [28]. The LR calculus deals with point type entities in the plane R^2 . It describes the position of a point C with respect to two other points A (the origin) and B (the relatum). However, in sketch maps landmarks are extended objects approximated by polygons and considering the centroids of landmarks loses shape information of the depicted objects.

We propose the following relations for representing relative orientation of landmarks and street segments at a conceptual level in sketch maps (see Figure 3). Our representation consists of six binary relations between two objects: *left_of*, *right_of*, *crosses*, *crossed_by*, *front_of*, and *back_of*. These relations capture key equivalence classes of spatial configurations that are preserved in sketch maps.

Orientation Relations. These rules define six orientation relations between depicted landmarks and street segments. A landmark is considered to be on the *left of* a street segment if all of the vertices of the polygonal representation of the landmark are on the left side of the street segment. The landmark *crosses* an adjacent street segment if some vertex of the landmark is *left_of* the street segment, some vertex is *right_of* the street segment, and the projection of the landmark

onto the line collinear with the street intersects the street segment.

```
orientation(left_of, polygon(Pts), line(A,B)) :-
  orientation_all_(left_of, Pts, line(A,B)).

orientation(crosses, polygon(Pts), line(A,B)) :-
  projection(PrjLm, polygon(Pts), line(A,B)),
  projection(PrjSt, segment(A,B), line(A,B)),
  not(allen(before, PrjSt, PrjLm)),
  not(allen(after, PrjSt, PrjLm)),
  orientation_some_(left_of, Pts, line(A,B)),
  orientation_some_(right_of, Pts, line(A,B)).
```

A landmark is in *front_of* an adjacent street segment if some vertex of the landmark is on the *left_of* the street segment, some vertex is on the *right_of* the street segment, and the projection of the landmark onto the line collinear with the street is not on the street segment (see Figure 3b). The respective inverse orientation relations are: *right_of*, *crossed_by* and *back_of*.

```
orientation(front_of, polygon(Pts), line(A,B)) :-
  orientation_some_(left_of, Pts, line(A,B)),
  orientation_some_(right_of, Pts, line(A,B)),
  projection(PrjLm, polygon(Pts), line(A,B)),
  projection(PrjSt, segment(A,B), line(A,B)),
  allen(before, PrjSt, PrjLm).
```

D. Rules for Cyclic Ordering as Constraints

This sketch aspect describes the angular ordering of nearby landmarks and connected street segments as referent objects around a street junction. Similar to linear ordering, Wang et al. [33, 34] find the cyclic ordering of both landmarks and street segments around reference junctions as a suitable representation for sketch map alignment.

In [16], we proposed a coarse version of the Cyclic Interval Algebra (CIA) [22] to formalize cyclic ordering of depicted sketch map objects around reference junctions. The c-intervals of landmarks and street segments are their projections onto a central point, i.e. a reference junction. The projection is given by sweeping the 360° view at a junction in a counter-clockwise direction. We define spatial rules in CLP(QS) for deriving the cyclic ordering of depicted objects. As with linear ordering, the spatial rule is used to generate a qualitative constraint network between landmarks and street segments based on adjacency, cyclic projections, and qualitative relations between cyclic intervals.

Adjacency Buffer Size. Analogous to the computation of buffer size around a street segment, this rule defines the buffer size around a reference object. The buffer size around junctions is used to define adjacency relations between landmarks and reference junctions.

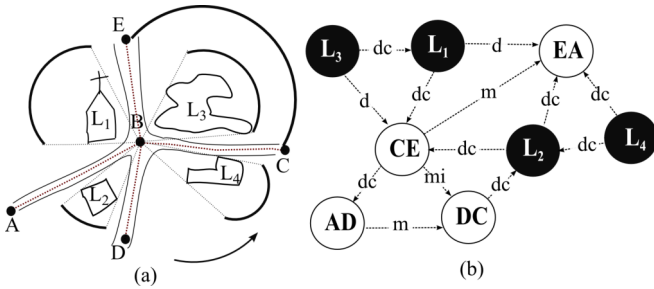


Fig. 4. (a) Cyclic ordering of adjacent landmarks with respect to junction B (b) qualitative constraints representing cyclic ordering relations

```

set_buffer_distance :-
    max_landmark_junction_distance(value(D)),
    retractall(buffer(_)), assert(buffer(D)).

max_landmark_junction_distance(value(D)) :-
    setof(Dist,
        L^nearest_junction_distance(value(Dist), L), Dists),
    sort(Dists, SDists), reverse(SDists, [D|_]).

nearest_junction_distance(value(MinD), L) :-
    object(L, type(landmark)), representation(L, LG),
    setof(D, S^SG^
        object(S, type(junction)),
        representation(S, SG),
        distance(value(D), LG, SG)
    ), Ds),
    sort(Ds, [MinD|_]).

```

Cyclic Projection. This rule projects the adjacent start and end points of landmarks onto reference junctions. The projected points represent c-intervals of adjacent landmarks. In order to further constrain the relative positions of landmarks around junctions, we also consider the c-intervals of *street segments* connected at the reference junctions. Each projected point in a c-interval represents a pair of connected street segments at a junction. The relations between these intervals represents the cyclic ordering of connected street segments.

```

cyclic_ordering(Rel, Lm1, Lm2, Jun) :-
    object(Jun, type(junction)),
    object(Lm1, type(landmark)),
    object(Lm2, type(landmark)),
    representation(Lm1, GLm1),
    representation(Lm2, GLm2),
    representation(Jun, GJun),
    cyclic_projection(PrjLm1, GLm1, GJun),
    cyclic_projection(PrjLm2, GLm2, GJun),
    cyclic_ordering(Rel, PrjLm1, PrjLm2).

```

Qualitative Cyclic Ordering. These rules define the cyclic ordering of landmarks and connected street segments using the projected c-intervals. It captures a panoramic view of objects around junctions in the form of cyclic ordering constraints. As illustrated in Figure 5, cyclic intervals are represented as a centre angle θ (measured counter-clockwise from the origin direction $(1,0)$) and an angular half-distance h between the centre and the end-points of the cyclic interval.¹ Figure 4 illustrates the *disconnected* (dc), *during* (d), *met-by* (mi), and *meets* (m) cyclic relations between two c-intervals of objects around a junction.

¹The half-distance of an interval is analogous to the radius of a circle being half of the circle's diameter.

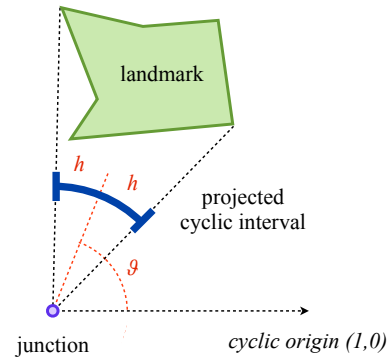


Fig. 5. Projected cyclic interval of a polygonal landmark onto a junction point. The cyclic interval is represented as a centre angle θ (relative to the angular origin $(1,0)$) and angular half-distance h .

```

cyclic_ordering(disconnected,
    cyclic_interval(centre(Ca), Ha),
    cyclic_interval(centre(Cb), Hb)) :-
    distance(radian(Dab),
        cyclic_interval(centre(Ca), Ha),
        cyclic_interval(centre(Cb), Hb)),
    {Dab > Ha + Hb}.

```

```

cyclic_ordering(externally_connects,
    cyclic_interval(centre(Ca), Ha),
    cyclic_interval(centre(Cb), Hb)) :-
    distance(radian(Dab),
        cyclic_interval(centre(Ca), Ha),
        cyclic_interval(centre(Cb), Hb)),
    {Dab == Ha + Hb}.

```

```

cyclic_ordering(partially_overlaps,
    cyclic_interval(centre(Ca), Ha),
    cyclic_interval(centre(Cb), Hb)) :-
    distance(radian(Dab),
        cyclic_interval(centre(Ca), Ha),
        cyclic_interval(centre(Cb), Hb)),
    {Dab < (Ha + Hb), Dab > abs(Ha - Hb)}.

```

```

cyclic_ordering(tangential_proper_part,
    cyclic_interval(centre(Ca), Ha),
    cyclic_interval(centre(Cb), Hb)) :-
    distance(radian(Dab),
        cyclic_interval(centre(Ca), Ha),
        cyclic_interval(centre(Cb), Hb)),
    {Dab == (Hb - Ha), Ha < Hb}.

```

```

cyclic_ordering(nontangential_proper_part,
    cyclic_interval(centre(Ca), Ha),
    cyclic_interval(centre(Cb), Hb)) :-
    distance(radian(Dab),
        cyclic_interval(centre(Ca), Ha),
        cyclic_interval(centre(Cb), Hb)),
    {Dab < (Hb - Ha), Ha < Hb}.

```

```

cyclic_ordering(meets,
    cyclic_interval(centre(Ca), Ha),
    cyclic_interval(centre(Cb), Hb)) :-
    cyclic_ordering(externally_connects,
        cyclic_interval(centre(Ca), Ha),
        cyclic_interval(centre(Cb), Hb)),
    {(Ca + Ha) modulo 2*pi == (Cb - Hb) modulo 2*pi}.

```

```

cyclic_ordering(met_by,
    cyclic_interval(centre(Ca), Ha),
    cyclic_interval(centre(Cb), Hb)) :-
    cyclic_ordering(meets,
        cyclic_interval(centre(Cb), Hb),
        cyclic_interval(centre(Ca), Ha)).

```

V. DISCUSSION AND USE CASE

In this section we discuss preliminary results of using our representations for matching. We have conducted a simple pilot study to investigate the ways in which various representations can be combined and utilised for the task of matching (i.e. aligning) spatial objects from sketch maps with geo-referenced maps.

The objective of the alignment task is to accurately match objects from the sketch map with objects in the geo-referenced map according to their relevant qualitative spatial relations. This is a challenging task in general [27], and we are investigating the use of a “mosaic” of different representations that can be combined to increase the accuracy of the matching process, e.g. a combination of topological relations between city blocks, linear representations between landmarks along street segments, cyclic representations between landmark around junctions, and so on.

In this simple pilot study we explore the combination of topological relations between city blocks and linear relations between landmarks based on a real, user-generated sketch map of Münster, Germany (see Figures 6 and 7). That is, we firstly match the maps based on city blocks using topological relations (primarily adjacency), as reported in [17]. As we report in [17], city blocks can often be matched accurately based on topological information. Figure 8 illustrates the extract of the sketch map (Figure 6) and geo-referenced map (Figure 7) comparing landmarks within a correctly matched city block.

We employ our linear representation to match landmarks within each city block. Landmarks are linearly ordered based on their projection onto the streets that define the city block. Figure 8 illustrates the derived ordering of the landmarks; for clarity, only the upper horizontal street in the sketch map is shown. The light grey region indicates the proximity threshold used to define adjacency of landmarks with respect to street segments. Projection of the landmark is based on the region of the landmark within the proximity threshold. Based on these projections, landmarks r_2, r_3, r_4 maintain a similar ordering in both the sketch and geo-referenced maps, thus assisting in the matching between landmark objects within a city block.

This pilots study also highlights the complexity of the matching task: observe that landmark r_1 was not sketched by the participant (i.e. r_1 is missing from the sketch map, Figure 8(a)). Moreover, according to our projection representation in the geo-referenced map (Figure 8(b)), r_1 overlaps r_2 in the first street of the geo-referenced map, but is also projected onto the last street that defines the block. We need to formalise further spatial domain knowledge in order for a matching algorithm to correctly interpret such combinations of qualitative relations between landmarks along a path.

The evaluation of proposed representations based on qualitative matching of sketch maps with geo-referenced maps is ongoing research work. In future work we are also investigating alternative landmark projections. For example, the front face of the polygon may be better indication of how people perceive the ordering information of landmarks along a particular route,



Fig. 6. Sketch map of a region in Münster, Germany.

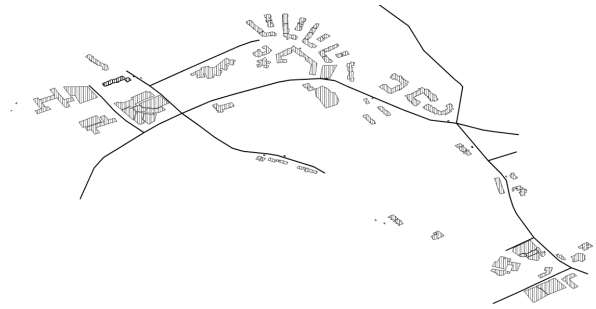


Fig. 7. Geo-referenced map of Münster, Germany.

rather than the projection of the entire polygon within the proximity threshold.

VI. CONCLUSIONS

In this paper we introduce a set of spatial relations that formalize key equivalence classes of spatial configurations preserved in sketch maps; the relations have been defined using the declarative spatial reasoning system CLP(QS). Our rules address: (1) the linear ordering of landmarks and street segments along any route; (2) the cyclic ordering of landmarks and connected street segments around reference junctions; (3) the orientation of landmarks with respect to nearby street segments. The rules also define an appropriate buffer size in sketch maps using the minimum distances between landmarks and street segments. The buffer size is used to define proximity and adjacency relations as qualitative relations between nearby objects.

We derive ordering and relative orientation relations as sets of Prolog facts using CLP(QS). We then use these facts to generate Qualitative Constraint Networks (QCNs) of sketch and geo-referenced maps. These qualitative networks are used in a range of tasks, for example (a) to provide a high-level qualitative query interface to both sketched and geo-referenced maps, and (b) to facilitate matching sketched maps to geo-referenced maps i.e. aligning qualitative spatial information in sketch maps with corresponding spatial information in geo-referenced maps. The evaluation of proposed representations based on qualitative matching of sketch maps with geo-referenced maps is ongoing research work.

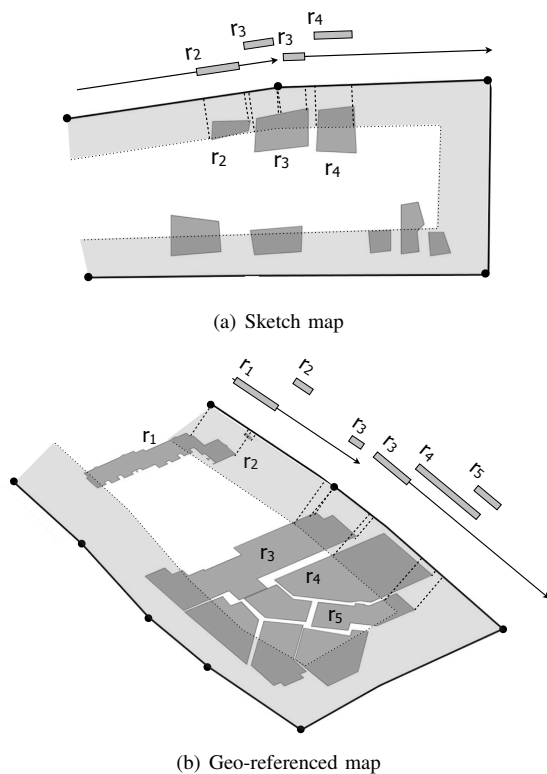


Fig. 8. Deriving linear ordering relations of projected landmarks per block in the sketch and geo-referenced map; only the projection of adjacent landmarks onto the upper horizontal street is illustrated.

ACKNOWLEDGMENT

This work is funded by the German Research Foundation (DFG) under grant for a SketchMapia project (Grant SCHW 1372/7-1).

REFERENCES

- [1] J. F. Allen. Maintaining Knowledge about Temporal Intervals. *Communications of the ACM*, 26(11):832–843, 1983.
- [2] M. Bhatt, J. H. Lee, and C. Schultz. CLP(QS): A Declarative Spatial Reasoning Framework. In *proceedings of the 10th international conference on Spatial information theory (COSIT-11)*, pages 210–230, Berlin Heidelberg, 2011. Springer-Verlag.
- [3] A. Blaser and M. J. Egenhofer. A visual tool for querying geographic databases. In *Proceedings of the working conference on Advanced visual interfaces*, number 1, pages 211–216, New York, New York, USA, 2000. ACM. ISBN 1581132522. doi: 10.1145/345513.345318.
- [4] K. Broelemann. A System for Automatic Localization and Recognition of Sketch Map Objects. In *Workshop of Understanding and Processing Sketch Maps*, pages 11–20, Belfast, Maine, 2011. AKA Verlag.
- [5] M. Chipofya, J. Wang, and A. Schwering. Towards cognitively plausible spatial representations for sketch map alignment. In *M. Egenhofer, N. Giudice, R. Moratz, M. Worboys (eds.) Spatial Information Theory, LNCS, Vol. 6899*, pages 20–39. Springer, Berlin Heidelberg, 2011.
- [6] A. G. Cohn. Qualitative Spatial Representation and Reasoning Techniques. In *Advances in Artificial Intelligence 1997. LNAI, Vol. 1303*, pages 1–30. Springer Berlin Heidelberg, 1997.
- [7] M. J. Egenhofer. Spatial-Query-by-Sketch. In *M. Burnett and W. Citrin (eds.), IEEE Symposium on Visual Languages (IEEE-96)*, Vol. 96, pages 60–67. IEEE Press, Boulder, Colorado, 1996.
- [8] M. J. Egenhofer. Query processing in spatial-query-by-sketch. In *the Journal of Visual Languages and Computing*, 8(4):403–424, 1997.
- [9] K. Forbus, J. Usher, A. Lovett, K. Lockwood, and J. Wetzel. CogSketch: Open-domain sketch understanding for cognitive science research and for education. volume 4, pages 648–666, 2011. doi: 10.1111/j.1756-8765.2011.01149.x.
- [10] A. U. Frank. Qualitative spatial reasoning: cardinal directions as an example. In *International journal of geographical information systems*, 10(3):269–290, Apr. 1996. ISSN 0269-3798. doi: 10.1080/02693799608902079.
- [11] C. Freksa. Dimensions of qualitative spatial reasoning. In *N.P. Carreté, M.G. Singh (eds.) Proceeding III MACS-International Workshop on Qualitative Reasoning and Decision Technologies QUARDET'93*, number June 1993, pages 483–492. CIMNE, Barcelona, 1993.
- [12] N. T. Huynh and S. T. Doherty. Digital Sketch-Map Drawing as an Instrument to Collect Data about Spatial Cognition. *Cartographica: The International Journal for Geographic Information and Geo-visualization*, 42(4):285–296, Jan. 2007. ISSN 0317-7173. doi: 10.3138/carto.42.4.285.
- [13] S. Jan and M. Chipofya. Integration of Qualitative Spatial Reasoning into GIS-An Example with SparQ. In *Schwering, A. Pebesma, E. Behncke, K. (eds.), In Geoinformatik 2011*, pages 63–78, Münster, Germany., 2011.
- [14] S. Jan, A. Schwering, M. Chipofya, and T. Binor. Qualitative Representations of Extended Spatial Objects in Sketch Maps. In *J. Huerta et al. (eds.), Connecting a Digital Europe Through Location and Place, LNG&C*, pages 37–54. Springer International Publishing, Switzerland, 2014. doi: 10.1007/978-3-319-03611-3_3.
- [15] S. Jan, A. Schwering, M. Chipofya, and J. Wang. Qualitative Representations of Schematized and Distorted Street Segments in Sketch Maps. In *International Conference, Spatial Cognition IX, LNCS, Vol. 8684*, pages 253–267, Bremen, Germany, 2014. Springer International. URL http://ifgi.uni-muenster.de/~s_jan001/Publications/sc2014.pdf.
- [16] S. Jan, A. Schwering, J. Wang, and M. Chipofya. Ordering: A Reliable Qualitative Information for the Alignment of Sketch and Metric Maps. In *International Journal of Cognitive Informatics and Natural Intelligence (IJCINI-2014)*, 8(1):68–79, 2014.
- [17] S. Jan, A. Schwering, C. Schultz, and M. Chipofya. RCC11: A formal topological representation for the alignment of Extended Objects in Sketch Maps. In *28th International Workshop on Qualitative Reasoning (QR-2015)*, August 10-12, Minneapolis, MN, USA, 2015.
- [18] K. Lynch. *The image of the city*. Cambridge MA, USA: MIT Press, 1960. ISBN 0262620014.
- [19] R. Moratz, J. Renz, and D. Wolter. Qualitative spatial reasoning about line segments. In *Proceedings of the 14th European Conference on Artificial Intelligence (ECAI 2000)*, pages 234–238. IOS Press, Amsterdam, 2000.
- [20] R. Moratz, F. Dylla, and L. Frommberger. A Relative Orientation Algebra with Adjustable Granularity. In *proceedings of the Workshop on Agents in Real-time and Dynamic Environments (IJCAI05)*, Edinburgh, Scotland, UK, 2005.
- [21] K. A. Nedas and M. J. Egenhofer. Spatial-Scene Similarity Queries. In *Transactions in GIS*, 12(6):661–681, 2008.
- [22] A. Osmani. Introduction to Reasoning about Cyclic Intervals. In *In Imam, I., Kodratoff, Y., El-Dessouki, A. and Ali, M. (Eds.) Multiple Approaches to Intelligent Systems. In Lecture Notes in Computer Science, Vol. 1611, Springer, Berlin-Heidelberg*, volume 1611, pages 698–706, 1999.
- [23] D. A. Randell, Z. Cui, and A. G. Cohn. A Spatial Logic based on Regions and Connection. In *B. Nebel, C. Rich, and W. R. Swartout (Eds.), In Proceedings of the 3rd National Conference on Principles of Knowledge Representation and Reasoning*, Cambridge, MA, 1992. Morgan Kaufmann Publishers Inc.
- [24] J. Renz and D. Mitra. Qualitative Direction Calculi with Arbitrary Granularity. In *C. Zhang, H. Guesgen, W. Yeap (eds.), PRICAI-04: Trends in Artificial Intelligence, LNCS, Vol. 3157*, pages 65–74. Springer, Berlin Heidelberg, Auckland, New Zealand, 2004. doi: 10.1007/978-3-540-28633-2_9.
- [25] C. Schlieder. Reasoning about ordering. In *Andrew U. Frank, Werner Kuhn (Eds.): Spatial Information Theory: A Theoretical Basis for GIS, International Conference COSIT '95, Semmering, Austria, September 21-23, 1995*.
- [26] C. Schultz and B. Mehul. Towards a Declarative Spatial Reasoning System. In *proceedings of the 20th European Conference on Artificial Intelligence (ECAI-12)*, Montpellier, France, 2012. ISO Press.
- [27] A. Schwering, J. Wang, M. Chipofya, S. Jan, R. Li, and K. Broelemann. SketchMapia: Qualitative Representations for the Alignment of Sketch and Metric Maps. *Spatial Cognition & Computation: An Interdisciplinary Journal*, 14(3):220–254, July 2014. ISSN 1387-5868. doi: 10.1080/13875868.2014.917378. URL <http://www.tandfonline.com/doi/abs/10.1080/13875868.2014.917378>.
- [28] A. Scivos and B. Nebel. The Finest of its Class: The Natural, Point-Based Ternary Calculus LR for Qualitative Spatial Reasoning. In *C. Freksa et al. (2005), Spatial Cognition IV. Reasoning, Action, Interaction: International Conference Spatial Cognition. Lecture Notes in Computer Science Vol. 3343, Springer, Berlin Heidelberg*, volume 3343, pages 283–303, 2004.
- [29] B. Tversky. Distortions in cognitive maps. In *Geoforum-interdisciplinary Journal*, 23(2):131–138, May 1992. ISSN 00167185. doi: 10.1016/0016-7185(92)90011-R.
- [30] B. Tversky. Structures of Mental Spaces: How People Think About Space. *Environment & Behavior*, 35(1):66–80, 2003.
- [31] H. Volker and W. Michael. Querying GIS with Animated Spatial Sketches. In *Proceeding IEEE Symposium on Visual Languages*, pages 197–204, Isle of Capri, Italy, 1997. IEEE Press.
- [32] J. Wallgrün, D. Wolter, and K.-F. Richter. Qualitative Matching of Spatial Information. In *proceedings of the 8th SIGSPATIAL International Conference on Advances in Geographic Information Systems*, pages 300–309, USA, 2010. ACM New York. ISBN 9781450304283.
- [33] J. Wang and A. Schwering. Invariant spatial information in sketch maps: Towards a sketching interface for collaborative mapping. In *Journal of Spatial Information Science*, 2014.
- [34] J. Wang, C. Mülligann, and A. Schwering. A Study on Empirically Relevant Aspects for Qualitative Alignment of Sketch Maps. In *proceedings of the Sixth international conference on Geographic Information Science (GIScience)*, 2010.

A Framework for Constructing Correct Qualitative Representations of Geometries using Mereology over Bintrees

Leif Harald Karlsen
Department of Informatics
University of Oslo
Email: leifhka@ifi.uio.no

Martin Giese
Department of Informatics
University of Oslo
Email: martingi@ifi.uio.no

Abstract—In this paper we explore how bintrees can function as a suitable representation for mereological objects, and how such objects can be used to construct correct representations of geometries, with respect to qualitative queries constructed from a given set of mereological relations. We will show how these correct representations can be stored and queried by a traditional relational database using relational algebra, or similar tuple-based databases.

We will define a model theoretic semantics for the bintrees and show how we can construct these correct representations as solutions to constraint networks with variables ranging over bintrees. Furthermore, we make an algorithm for solving the constraints and prove its correctness.

The framework presented in the paper is not limited to only constructing representations of geometries, but representations of any objects where a *part-of* relationship is natural.

I. INTRODUCTION

GEOSPATIAL and temporal data is ubiquitous in today's software, with a growing number of spatially aware devices gathering and publishing data. Spatial and temporal data is used in a great number of highly valuable applications, like route planning, automatic navigation, modelling of physical processes, etc. However, temporal and especially geospatial data are normally represented as complex numerical objects that are difficult to represent in information storing software. The relationships between objects are in these numerical representation implicit, and one needs advanced numerical algorithms for exacting this data. Storing such data with regards to efficient information retrieval is also difficult, as indexing these objects are far from easy.

During the last decades, several temporal and geospatial database systems have been developed, featuring advanced indexing mechanisms and efficient numerical algorithms for answering queries over such data (see e.g. [1]). Despite these advances, geospatial and temporal data is still a lot more difficult to handle than more traditional data. These data types also often lag behind when new knowledge representations are introduced and often need special treatment.

We therefore want to create a framework for constructing non-numerical representations of numerically represented geometries (and other numerically represented elements). These

representations should be in a format that we can store and query in a relational database and other tuple based storage structures (e.g. a triple store) where properties of the elements are stored explicitly. In addition to qualitative query answering, we also want to be able to pose window queries, that is queries involving geometrical constants, and have efficient insert of new objects.

The queries we want our system to handle are qualitative queries, that is, queries that consist of only non-numerical predicates (e.g. *Overlaps(a,b)* and *Contains(a,b)*). Such relationships tend to be closer to how humans generally think, and are often sufficient for a geospatial database. These relations are naturally expressed using the *part-of* relation (see example 11). The part-of relation is the base relation in the theory of mereology in the same way the element-in relation is the base relation of set theory. We will therefore base our framework around mereological queries.

Our approach will solve the problem above by using a type of geospatial index structures called linear bintrees [2], which is a type of trie. Each node in the tree represents an area, with the root node representing some universe. Every node has two children, each representing half of the area of its parent node. Furthermore, every node is denoted by a bit-string. The root node is represented with the empty bit-string, and each left and right child-node of a node is represented with its parent bit-string s but with a 0 or a 1 appended to the end of s respectively. A spatial object can then be represented as a union (i.e. as a set) of bit-strings, which then represents the union of the areas each node with the given bit-strings represents.

Bintrees index geometries by constructing a set of nodes from the bintree that represents an approximation (from above) of the area of each geometry. Such an index structure can then be used to quickly compute a complete (but not sound) approximation of the answers to a query, that can then be filtered by using the actual geometries. Bintrees have the convenient property that they can be stored as a regular database relation, and indexed by normal database index structures, like B-trees, since they only consists of sets of bit-strings. Another nice feature of bintrees is that they allow variable resolution, so we

can have low resolution (small bit-strings) for homogeneous areas and high resolution (long bit-strings) for homogeneous areas where more detail is necessary.

We will use bintrees as a representation, but fix the approximation by constructing mereological constraints over the bintrees, such that any solution will give correct representations with respect to mereological queries. The framework we develop throughout this paper is not restricted to only geometries, but can be used for any type of data that satisfies the axioms of Classical extensional mereology (see Definition 6).

The paper is organised as follows. Section

- II gives a naive approach and explains why this is unfeasible, and why using an index structure is desirable;
- III properly defines the bintrees, the mereological relation \prec and the models we will be using throughout this paper;
- IV introduces the relations we will allow our queries over the representations to contain;
- V defines how our mereological models can be interpreted geometrically, and defines what *correct* representations are with respect to the geometries;
- VI introduces mereological constraints that we can use to state the properties we want our representations to have. The constraints is our main tool for constructing correct representations;
- VII explains how we can construct constraints that properly describe correctness, and we will here define functions for automatic construction of such constraints;
- VIII describes an algorithm for solving the constraints, and contains a proof of correctness of the solver;
- IX outlines the details of query answering over our representation, and how we can answer mereological queries in relational algebra, Datalog, and SPARQL;
- X examines the time complexity of the solver and the bounds of the storage space of the representations;
- XI contains a summary of the results, and ideas for extensions of the framework.

Throughout this paper, we will assume that \mathcal{G} is a finite set of names of objects that we want to represent. They can either be temporal, spatial, or other types of objects for which part-of relationships are natural. We put no restriction on the number of dimensions these elements have, as long as they all have the same (finite) number of dimensions.

II. A NAIVE APPROACH AND THE OUTLINE OF A SOLUTION

Assuming we have a numerical representation of the elements we want to describe, a naive approach to the problem could be to simply construct a table for each relation, compute all possible relations between the elements and store the tuples in the tables. However, this would require storing tables with an exponential number of tuples in each. Furthermore, if we do not have an index structure, every time you insert a new element you will have to compute all relationships between the new element and every geometry already in the database. Furthermore, such a solution would not allow us to pose window queries.

Hence, a feasible solution needs a spatial index structure that allows us quickly to look up which objects are potentially spatially related to the new element. Such a structure could then also be used for posing window queries, as it gives an approximate location of each object.

Index structures are complete, so the elements returned from a look-up for a query q should contain at least all the answers to q . However, spatial index structures are often not sound. The main idea behind our approach is to construct sound and complete index structures, such that we only have to make an index look-up to answer a query.

III. MEREOLOGY OVER BINTREES

As we saw from previous section, we will use a spatial index structure for representing the objects of \mathcal{G} . As stated in the introduction, a data structure that is well suited for our use is the linear bintree [2] index structure.

In this section we will properly define this data structure, and see how the mereological part-of relation, denoted \prec , can be defined over bintrees.

Definition 1. *Define*

- \mathcal{B} to be the set of bit-strings called blocks, and where ε is the empty bit-string (that is $\mathcal{B} = \{0, 1\}^*$);
- $s \circ s'$ to be the concatenation of the bit-strings s and s' from \mathcal{B} , with ε as identity;
- $s_1 \triangleleft s_2 \Leftrightarrow \exists s(s_2 \circ s = s_1)$, that is, $s_1 \triangleleft s_2$ states that the block s_2 is a (string) prefix of the block s_1 ;
- two blocks $s_1, s_2 \in \mathcal{B}$ to be neighbours if there exists an $s \in \mathcal{B}$ s.t. $s_1 = s \circ 0$ and $s_2 = s \circ 1$.

Since the prefix relation on strings is a partial order, so is \triangleleft .

Definition 2. *Define the set of bintrees \mathcal{M} be the set of $\alpha \in \mathcal{P}_{fin}(\mathcal{B}) \setminus \{\emptyset\}$ (where \mathcal{P}_{fin} is the set of finite subsets) such that α contains no neighbours and no two (unequal) elements s_1, s_2 where $s_1 \triangleleft s_2$.*

Furthermore, define the depth of an element $\alpha \in \mathcal{M}$ to be the length of the longest bit-string in α , denoted $\Delta(\alpha)$.

The set \mathcal{B} will represent the blocks in a bintree, while each $\alpha \in \mathcal{M}$ will be a set of such blocks, representing an area. Notice that there is no assumption on the number of dimensions in the representation (except for finiteness). Notice that disallowing neighbours and pairs of \triangleleft -related elements gives us the optimal representation for each area.

We will need a formal framework for studying the properties of different representations. First order logic with model theoretic semantics is a suitable language for studying such properties. The models we are going to work with and the model semantics we will use is defined below.

Definition 3. *Let a mereological model \mathcal{Q} be a first order model for the language with a binary relation symbol \prec , and constants $\mathcal{G} \cup \mathcal{M}$, such that*

- (i) *the model's universe is \mathcal{M} ;*
- (ii) *$a^{\mathcal{Q}} = a$ for any $a \in \mathcal{M}$, i.e. bintree literals are interpreted as themselves; and*

(iii) $\prec^Q = \prec$ where

$$a \prec b \Leftrightarrow \forall s \in a \exists s' \in b (s \prec s')$$

The interpretation of the constants in \mathcal{G} is not constrained.

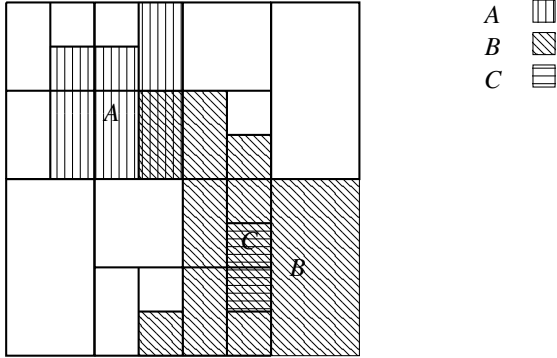
Note that the only difference between two mereological models is their interpretations of the constants in \mathcal{G} , everything else is fixed. Since our models are just special instances of first order models we will use the regular notation of first order logic, such as the satisfaction relation \models and first order formulas.

Definition 4. Let $\models_{\mathcal{M}}$ be the mereological consequence relation, such that $\varphi \models_{\mathcal{M}} \psi$ holds iff for any mereological model Q we have $Q \models \varphi \Rightarrow Q \models \psi$.

Throughout this article we will often use the relation $O(a, b) \Leftrightarrow \exists v (v \prec a \wedge v \prec b)$, which is the *overlaps* relation. We will sometimes abuse notation and state the truth value of $O(a, b)$ outside a model when $a, b \in \mathcal{M}$. In these cases we will assume that by $O(a, b)$ we mean $\exists v \in \mathcal{M} (v \prec a \wedge v \prec b)$.

We will also use the shorthand $a \not\prec b$ and $a \dot{\prec} b$ instead of $\neg(a \prec b)$ and $\neg(a \dot{\prec} b)$ respectively.

Example 5. Let's construct a toy example with three two-dimensional areas, so let $\mathcal{G} = \{A, B, C\}$.



The above image is a visualisation of the model Q where

$$\begin{aligned} A^Q &= \{000011, 00011, 0011, 00101, 001001\}, \\ B^Q &= \{00111, 10010, 1001011, 11, 011111\}, \\ C^Q &= \{110011, 110110\} \end{aligned}$$

We can see that $Q \models \exists v (v \prec A \wedge v \prec B)$ (since $\{00111\}$ is part of both) and $Q \models C \prec B$.

We will now take a brief look at the mereological system our definitions satisfy.

Definition 6. Classical extensional mereology [3] (CEM) has the following axioms for \prec :

- (i) *Reflexive:*
 $\forall x (x \prec x)$;
- (ii) *Transitive:*
 $\forall x \forall y \forall z (x \prec y \wedge y \prec z \rightarrow x \prec z)$;
- (iii) *Anti-symmetric:*
 $\forall x \forall y (x \prec y \wedge y \prec x \rightarrow x = y)$;

(iv) *Top:*

$$\exists y \forall x (x \prec y);$$

(v) *Strong supplementation:*

$$\forall x \forall y (y \not\prec x \rightarrow \exists z (z \prec y \wedge \neg O(z, x)));$$

(vi) *Sum:*

$$\forall x \forall y \exists z \forall v (O(v, z) \leftrightarrow (O(v, x) \vee O(v, y)));$$

(vii) *Product:*

$$\forall x \forall y (O(x, y) \rightarrow \exists z \forall v (v \prec z \leftrightarrow (v \prec x \wedge v \prec y))).$$

The framework we will construct in this paper will be able to construct correct representations with respect to any relation constructed from a base relation satisfying the axioms of Definition 6.

Lemma 7. Our definition of \prec over bintrees satisfies the axioms for CEM.

Proof. We will use the same enumeration of the axioms as in Definition 6:

- (i) *Reflexivity:* Follows easily from the reflexivity of \prec .
- (ii) *Transitivity:* Assume $\forall s \in x \exists s' \in y (s \prec s')$ and $\forall s' \in y \exists s'' \in z (s' \prec s'')$, then we have that $\forall s \in x \exists s' \in y \exists s'' \in z (s \prec s' \wedge s' \prec s'')$. By transitivity of \prec , the result follows.
- (iii) *Anti-symmetric:* If $\forall s \in x \exists s' \in y (s \prec s')$ and $\forall s' \in y \exists s \in x (s' \prec s)$, we must have that $x = y$ (i.e. they contain exactly the same elements from \mathcal{B}), since, by construction of \mathcal{M} , no element can contain two blocks s_1, s_2 where $s_1 \prec s_2$.
- (iv) *Top:* We have $\{\epsilon\}$ which satisfies this.
- (v) *Strong supplementation:* Assume $x, y \in \mathcal{M}$ where $y \not\prec x$. Then we have there must be one $s \in y$ where $\forall s' \in x (\neg s \prec s')$. By definition, there are no neighbours in y , it must be the case that there is an $s'' \prec s$ s.t. $\forall s' \in x (\neg s \prec s' \wedge \neg s' \prec s)$. We then have that $\{s''\} \prec y \wedge \neg O(\{s''\}, x)$.
- (vi) *Sum:* Assume $x, y \in \mathcal{M}$ and let $z' := x \cup y$. Let z be the element that results from recursively merging all neighbours of z' and removing all blocks s where $s \prec s'$ and $s, s' \in z'$. It should be easy to see that for any $v \in \mathcal{M}$, $O(v, z)$ holds iff $O(v, x)$ or $O(v, y)$.
- (vii) *Product:* Assume $x, y \in \mathcal{M}$ and $O(x, y)$. Let $z := \{s \in \mathcal{B} \mid (s \in x \wedge \{s\} \prec y) \vee (s \in y \wedge \{s\} \prec x)\}$. It should be easy to see that for any $v \in \mathcal{M}$ we have that $v \prec z$ iff $v \prec x$ and $v \prec y$.

□

From the above lemma, we can see that our definition of \prec satisfies the axioms of the theory CEM. Since bintrees are easy to store and index they are a natural choice for representing mereological objects on a computer.

The three last axioms of CEM states the existence of a difference, sum and product respectively. In the following lemma we should that the corresponding operators are well defined.

Lemma 8. We have that for any $x, y \in \mathcal{M}$:

- (i) if $y \not\prec x$ there is a unique \prec -maximal element $z \in \mathcal{M}$ satisfying $z \prec y \wedge \neg O(z, x)$;

- (ii) there is a unique element $z \in \mathcal{M}$ satisfying $\forall v \in \mathcal{M}(O(v, z) \leftrightarrow (O(v, x) \vee O(v, y)))$;
- (iii) if $O(x, y)$ there is a unique element $z \in \mathcal{M}$ satisfying $\forall v \in \mathcal{M}(v \prec z \leftrightarrow (v \prec x \wedge v \prec y))$.

Proof. We know from Lemma 7 that such objects exist in \mathcal{M} , we will prove that they are unique.

- (i) Let z be the set of \prec -greatest elements $s \in \mathcal{B}$ that satisfy $\{s\} \prec y \wedge \neg O(\{s\}, x)$. It should be easy to see that $z \in \mathcal{M}$ and that $z \prec y \wedge \neg O(z, x)$. Assume that there is an element $z' \in \mathcal{M}$ that satisfies $z' \prec y \wedge \neg O(z', x)$. For any $s \in z'$ we must have $\{s\} \prec y \wedge \neg O(\{s\}, x)$. Then, by definition of z , we have $\{s\} \prec z$. Since s was arbitrary $z' \prec z$, and since z' was arbitrary z must be the \prec -maximum, thus unique and maximal.
- (ii) Assume, for the sake of contradiction, that there are two unequal elements $z, z' \in \mathcal{M}$ that both satisfies the sentence. Then it follows that $\forall v(O(v, z) \leftrightarrow O(v, z'))$. Since z and z' are unequal there must exist an $s \in \mathcal{B}$ s.t. either $\{s\} \prec z \wedge \neg O(\{s\}, z')$ or $\{s\} \prec z' \wedge \neg O(\{s\}, z)$. However, both contradicts $\forall v(O(v, z) \leftrightarrow O(v, z'))$, so there is only one element $z \in \mathcal{M}$ satisfying the sentence.
- (iii) Assume that we have two elements $z, z' \in \mathcal{M}$ satisfying the sentence. Then $\forall v(v \prec z \leftrightarrow v \prec z')$ which implies $z = z'$.

□

Lemma 8 guarantees that the notions in the following definition are well-defined:

Definition 9. Let $x, y \in \mathcal{M}$.

- (i) If $y \not\prec x$, we will write $x \ominus y$ to denote the unique \prec -maximal element $z \in \mathcal{M}$ that satisfies $z \prec y \wedge \neg O(z, x)$. We call $x \ominus y$ the difference between x and y .
- (ii) We will write $x \oplus y$ to denote the unique element $z \in \mathcal{M}$ that satisfies $\forall v \in \mathcal{M}(O(v, z) \leftrightarrow (O(v, x) \vee O(v, y)))$. We call $x \oplus y$ the sum or the union of x and y .
- (iii) If $O(x, y)$, we will write $x \otimes y$ to denote the unique element $z \in \mathcal{M}$ that satisfies $\forall v \in \mathcal{M}(v \prec z \leftrightarrow (v \prec x \wedge v \prec y))$. We call $x \otimes y$ the product or intersection of x and y .

In the next section we will define which queries we will be able to answer over our representations.

IV. MERELOGICAL RELATIONS

As we saw in the previous section, bintrees represents mereological objects in a natural way. Our main reason for using bintrees is to ease storage and retrieval of information in a relational or similar tuple-based database. To this end, we will now introduce the query language we will use over our structures. It is well known that conjunctive queries have nice computational properties and are well supported over most tuple-based databases [4]. Our query language will therefore have base relations that are conjunctive queries.

Definition 10. Let \mathcal{V} be a set of variables, disjoint from \mathcal{G} . A mereological formula is a formula on the form defined by the BNF grammar:

$$\varphi ::= \varphi_1 \wedge \varphi_2 \mid \exists z . \varphi_1 \mid \alpha \prec \beta$$

where $z \in \mathcal{V}$ a variable, and $\alpha, \beta \in \mathcal{V} \cup \mathcal{M} \cup \mathcal{G}$. A mereological relation is an n -ary relation on \mathcal{M} described by a mereological formula. We will write r_φ for the mereological relation described by the mereological formula φ . Let the set of mereological relations be denoted $\mathcal{R}_\mathcal{M}$.

Furthermore, we will call a mereological formula with no free variables a mereological sentence. Whenever we write $\varphi(\vec{p})$ for a mereological formula φ and a vector \vec{p} of elements from $\mathcal{G} \cup \mathcal{M}$, we will assume that it is a mereological sentence (so the length of \vec{p} is equal to the number of free variables in φ .)

Example 11. Below is a list of examples of common relations expressed as mereological formulas:

- $Overlaps_2(p_1, p_2) = \exists z(z \prec p_1 \wedge z \prec p_2)$,
- $Overlaps_n(p_1, \dots, p_n) = \exists z(z \prec p_1 \wedge \dots \wedge z \prec p_n)$,
- $Contains(p_1, p_2) = p_1 \prec p_2$,
- $Between(p_1, p_2, p_3) = p_1 \prec p_2 \wedge p_2 \prec p_3$,
- $InIntersection(p_1, p_2, p_3) = p_3 \prec p_1 \wedge p_3 \prec p_2$,
- $Underlaps(p_1, p_2, p_3) = p_1 \prec p_3 \wedge p_2 \prec p_3$.

The set of mereological relations might seem inexpressive, but note that these are only base relations. When we know that the mereological objects are correctly represented according to these base relations, we can then use those relations to form more complex relations in a more expressive query language (e.g. SQL).

Example 12. Below we have defined the RCC5-relations [5] in terms of the base relations in the previous example:

- $DC(p_1, p_2) = \neg Overlaps_2(p_1, p_2)$,
- $O'(p_1, p_2) = Overlaps_2(p_1, p_2) \wedge \neg Contains(p_1, p_2) \wedge \neg Contains(p_2, p_1)$,
- $PP(p_1, p_2) = Contains(p_2, p_1) \wedge \neg Contains(p_1, p_2)$,
- $PP^{-1}(p_1, p_2) = PP(p_2, p_1)$,
- $EQ(p_1, p_2) = Contains(p_1, p_2) \wedge Contains(p_2, p_1)$.

V. MODELS OF GEOMETRY

To construct correct representations of the geometrical objects in \mathcal{G} , we will need geometrical models that interprets the elements of \mathcal{G} and \mathcal{M} as geometrical objects. We will therefore construct models over the same language, but with a different domain; the domain of sets of points in \mathbb{R}^d (for some finite number of dimensions d).

Definition 13. Assume that d denotes the (finite) number of dimensions we are working in. Let a geometrical model N be a first order model for the language with a binary relation symbol \prec , and constants $\mathcal{G} \cup \mathcal{M}$, such that

- (i) the model's universe is $\mathcal{N} := \mathcal{P}(\mathbb{R}^d)$;

(ii) the elements of \mathcal{M} are interpreted to elements of \mathcal{N} such that the following holds:

$$\begin{aligned} \{(s \circ 0)\}^N \cup \{(s \circ 1)\}^N &= \{s\}^N \text{ for any } \{s\} \in \mathcal{M}, \\ \{(s \circ 0)\}^N \cap \{(s \circ 1)\}^N &\subseteq \emptyset \text{ for any } \{s\} \in \mathcal{M}, \\ \alpha^N &= \bigcup_{s_i \in \alpha} \{s_i\}^N \text{ for any } \alpha = \{s_1, s_2, \dots, s_n\}; \end{aligned}$$

(iii) $\prec^N = \subseteq$.

The interpretation of the constants in \mathcal{G} is not constrained.

While mereological models differ only in their interpretation of the constants from \mathcal{G} , geometrical models can differ also in their interpretation of bintrees from \mathcal{M} to point sets in \mathcal{N} , subject to the constraints given in (ii). We will use standard first order logic syntax also for our geometrical models.

Definition 14. Let $\models_{\mathcal{N}}$ be the geometrical consequence relation, such that $\varphi \models_{\mathcal{N}} \psi$ holds iff for all geometrical models N we have $N \models \varphi \Rightarrow N \models \psi$.

Theorem 15. For any two mereological sentences φ, γ , we have

$$\varphi \models_{\mathcal{M}} \gamma \Leftrightarrow \varphi \models_{\mathcal{N}} \gamma$$

Proof. By the deduction theorem of first order logic, it suffices to prove $\models_{\mathcal{M}} \varphi \rightarrow \gamma \Leftrightarrow \models_{\mathcal{N}} \varphi \rightarrow \gamma$. We will prove that $\models_{\mathcal{M}} \alpha \prec \beta \Leftrightarrow \models_{\mathcal{N}} \alpha \prec \beta$ for $\alpha, \beta \in \mathcal{M}$, and the rest follows by standard first order logic.

By property (ii) of Definition 13 (and an easy induction proof), we have that $\models_{\mathcal{N}} \{s\} \prec \{s'\}$ if $s \prec s'$. If we combine this with property (iii) (and an easy induction proof), we have $\{s\}^N \subseteq \{s'\}^N$ only if $s \prec s'$, since $\{s \circ 0\} \cup \{s \circ 1\}$ partitions $\{s\}$. Then, by standard set theory and property (v), we can conclude that $\models_{\mathcal{N}} \bigcup_i \{s_i\} \prec \bigcup_j \{s'_j\} \Leftrightarrow \forall s_i \exists s'_j (s_i \prec s'_j)$. \square

We are now ready to state what we mean with correct representations.

Definition 16. Given a set of relations $R \subseteq \mathcal{R}_{\mathcal{M}}$, we say that a model Q is R -complete with respect to a geometrical model N if for any mereological relation $r_{\varphi} \in R$ and any tuple $\vec{p} \in \mathcal{G}^*$, we have

$$N \models \varphi(\vec{p}) \Rightarrow Q \models \varphi(\vec{p})$$

We say that Q is R -sound with respect to N if for any mereological relation $r_{\varphi} \in R$ and any tuple $\vec{p} \in \mathcal{G}^*$ we have

$$Q \models \varphi(\vec{p}) \Rightarrow N \models \varphi(\vec{p})$$

We want to construct a model Q such that it properly represents the geometries of \mathcal{G} according to a set of relations $R \subseteq \mathcal{R}_{\mathcal{M}}$, that is, it should be both R -sound and R -complete. As stated earlier, we also need our representations to function as a spatial index structure. For this we must be able to determine which objects are spatially related to an upper approximation of an object. Bintrees as spatial index structures normally have a maximum depth δ that decides the resolution of the approximation.

Definition 17. Assume that δ is a natural number denoting some initial maximal depth and let $\mathcal{M}_{\delta} := \{\alpha \in \mathcal{M} \mid \Delta(\alpha) \leq \delta\}$ be the set of elements of \mathcal{M} with depth less than or equal to δ . Let a mereological unary relation over \mathcal{M} be called a δ -index relation if it is described by one of the formulas $\beta \prec v$, $v \prec \beta$ or $O(v, \beta)$ for some $\beta \in \mathcal{M}_{\delta}$ and $v \in \mathcal{V}$. Define R_{δ} to be the set of δ -index relations.

The relations of R_{δ} can describe an object correctly up to resolution δ . In other words, any mereological model that correctly represents the elements of \mathcal{G} according to R_{δ} with respect to a geometrical model N , will function as a spatial index at the depth δ . Therefore, a mereological model that satisfies the the same sentences constructed from the relations of both R and R_{δ} as some geometrical model N , will be a sound and complete index structure.

In the next section we will introduce mereological constraints. These constraints will allow us to state what properties the elements of \mathcal{G} should have. In section VII we will see how we can use a geometrical model to construct constraints that will make any model of a solution function as a sound and complete index structure.

VI. MEREOLOGICAL CONSTRAINTS

With both a data structure to represent our objects and a query language over them, we can now talk about how we will construct our representations. To this end, it is natural to be able to state the properties we want our representations to have, and then find a representation that satisfies those properties. If we view the properties as constraints, the process of finding a proper representation would then be constraint solving.

We will now introduce mereological constraints, that is, constraints that express mereological relations between mereological objects.

Definition 18. Assume ψ is a mereological formula. Define $\mathcal{V}(\psi)$ to be the set of variables $v \in \mathcal{V}$ in ψ , $\mathcal{G}(\psi)$ to be the set of elements from \mathcal{G} in ψ , $\mathcal{M}(\psi)$ to be the set of elements from \mathcal{M} in ψ . Set $\mathcal{GV}(\psi) = \mathcal{G}(\psi) \cup \mathcal{V}(\psi)$ and $\mathcal{E}(\psi) = \mathcal{GV}(\psi) \cup \mathcal{M}(\psi)$.

Furthermore, a quantifier-free mereological formula ψ is a constraint if $\mathcal{G}(\psi)$ is nonempty.

By definition a constraint is any formula of the form $\bigwedge_i \alpha_i \prec \beta_i$, where $\alpha_i, \beta_i \in \mathcal{V} \cup \mathcal{G} \cup \mathcal{M}$. A constraint is therefore a formula that constrains the possible interpretations of the elements of \mathcal{G} . Note that even though a constraint is only one formula, it can be a large conjunction, and therefore constrain many or all of the elements of \mathcal{G} .

We will, in the rest of the paper, in addition to treating constraints as formulas, treat constraints both as a graph of \prec -edges, and a set of conjuncts. We will also abuse notation and write $(\alpha \prec \beta) \in \psi$ to mean that $\alpha \prec \beta$ is a conjunct in ψ .

Definition 19. A solution to a constraint ψ is a function $\sigma : \mathcal{GV}(\psi) \rightarrow \mathcal{M}$ such that the formula ψ' resulting from

substituting each free variable v in ψ with $\sigma(v)$, denoted $\psi\sigma$, is valid.

Since the domain of σ is $\mathcal{GV}(\psi)$, there will only be elements of \mathcal{M} in $\psi\sigma$. A substitution σ can therefore be verified as a solution without consulting any models or doing any reasoning except for computing \prec -relationships over constants from \mathcal{M} .

Definition 20. An interpretation Q is a model of a constraint ψ if σ is a solution to ψ and $\alpha^Q = \sigma(\alpha)$ for any $\alpha \in \mathcal{G}(\psi)$.

Every model of a constraint ψ must agree with some solution of ψ on the interpretations of the elements of $\mathcal{G}(\psi)$.

Definition 21. We say that a sentence φ is entailed by a constraint ψ and write $\psi \models_{\mathcal{M}} \varphi$, if for every model Q of ψ we have $Q \models \varphi$.

Our relations are going to be evaluated in a specific model that interprets the elements of \mathcal{G} . Because of this, our queries will be evaluated under the closed world assumption. This assumption is common to use in relational database systems and states that anything that is not known (read *derivable* or *entailed*) to be true, is false [4]. It is therefore essential for our solutions to be solved under the closed world assumption. Our solutions should therefore induce models that only satisfy the sentences entailed by the constraints. In other words, we want the minimal models of the constraints.

Definition 22. A model Q is minimal for a constraint ψ if for any mereological sentence φ , such that $\mathcal{G}(\varphi) \cup \mathcal{M}(\varphi) \subseteq \mathcal{G}(\psi) \cup \mathcal{M}(\psi)$, we have

$$Q \models \varphi \Leftrightarrow \psi \models_{\mathcal{M}} \varphi$$

A minimal model is then a model that satisfies exactly the same sentences as the constraints, if we limit the constants (both from \mathcal{G} and \mathcal{M}) to those of the constraints.

Definition 23. Assume that $\mathcal{G}(\psi) = \mathcal{G}$ for some constraint ψ . We say that Q is induced by a solution σ of ψ if $p^Q = \sigma(p)$ for all $p \in \mathcal{G}$.

Example 24. Assume we have $\mathcal{G} = \{A, B, C\}$ and that

$$\begin{aligned} \psi := & \{0011, 0110\} \prec A \wedge A \prec \{0\} \wedge A \prec B \wedge \\ & B \prec \{0\} \wedge \{100, 11\} \prec C \wedge v \prec C \wedge v \prec B \end{aligned}$$

We now have that e.g. $\psi \models_{\mathcal{M}} A \prec B \wedge O(B, C)$, but $\psi \not\models_{\mathcal{M}} O(A, C)$. A possible solution σ_1 could be

$$\begin{aligned} \sigma_1(A) &:= \{0011, 011\} & \sigma_1(B) &:= \{0\} \\ \sigma_1(C) &:= \{100, 11, 01\} & \sigma_1(v) &:= \{01\} \end{aligned}$$

It is a solution, since

$$\begin{aligned} \psi\sigma_1 = & \{0011, 0110\} \prec \{0011, 011\} \wedge \{0011, 011\} \prec \{0\} \wedge \\ & \{0011, 011\} \prec \{0\} \wedge \{0\} \prec \{0\} \wedge \\ & \{100, 11\} \prec \{100, 11, 01\} \wedge \{01\} \prec \{100, 11, 01\} \wedge \\ & \{01\} \prec \{0\} \end{aligned}$$

is valid. However, a model induced by σ_1 is not minimal, since $O(\sigma_1(A), \sigma_1(C))$ and $\{0\} \prec \sigma_1(B)$, neither of which

are entailed by ψ . The following modified solution induces a minimal model:

$$\begin{aligned} \sigma_2(A) &:= \{0011, 011\} & \sigma_2(B) &:= \{00, 011\} \\ \sigma_2(C) &:= \{100, 11, 000\} & \sigma_2(v) &:= \{000\} \end{aligned}$$

It is naturally important to know when it is possible to find a solution to a constraint, that is, when a constraint is consistent. Before we can define consistency of our constraints, we need a couple of important, albeit technical, definitions.

Definition 25. Assume ψ is a constraint. Let β, β' be called a c-pair in ψ if $\beta, \beta' \in \mathcal{M}(\psi)$ and $\beta \prec \beta'$. Let ψ_c be $\psi \cup \{\beta \prec \beta' \mid \beta, \beta' \text{ a c-pair in } \psi\}$. Let ψ^* be the transitive, reflexive closure of ψ_c with respect to \prec .

So ψ^* extends ψ with all implicit \prec -relationships that we have in ψ .

Definition 26. Assume ψ is a constraint and $\alpha \in \mathcal{E}(\psi)$. Define $R_{\prec}^{\psi}(\alpha) := \{\beta \in \mathcal{E}(\psi) \mid (\beta \prec \alpha) \in \psi^*\}$ and $R_{\succ}^{\psi}(\alpha) := \{\beta \in \mathcal{E}(\psi) \mid (\alpha \prec \beta) \in \psi^*\}$. We will call the elements of $R_{\prec}^{\psi}(\alpha)$ the \prec -successors of α and the elements of $R_{\succ}^{\psi}(\alpha)$ the \prec -predecessors of α .

$R_{\prec}^{\psi}(\alpha)$ contains all elements that are constrained to be a part of α in the constraints ψ , and R_{\succ}^{ψ} contains all elements that is constrained to have α as a part.

Definition 27. Let ψ be a constraint and $\alpha \in \mathcal{GV}(\psi)$. Define

$$M(\alpha) := \bigotimes_{\beta \in R_{\prec}^{\psi}(\alpha) \cap \mathcal{M}(\psi)} \beta$$

if R_{\prec}^{ψ} is nonempty, and $\{\epsilon\}$ otherwise.

$M(\alpha)$ is the element of \mathcal{M} which α is bound to be a part of, that is, it is the upper bound for any solution of α . Note that the only way $M(\alpha)$ can be undefined, is if we have a constraint where an α has two non-overlapping \prec -successors. If this is not the case, it should be easy to see (by looking at ψ as a graph of \prec -edges) that we can set $M(\alpha)$ to be equal to the intersection of all \prec -successors that are in \mathcal{M} .

Definition 28. Let ψ be a constraint and $\alpha \in \mathcal{GV}(\psi)$. Define

$$m(\alpha) := \bigoplus_{\beta \in R_{\succ}^{\psi}(\alpha) \cap \mathcal{M}(\psi)} \beta$$

$m(\alpha)$ is the element of \mathcal{M} which is bound to be a part of α , that is, the lower limit of any solution to α . If a constraint has an element α that does not have any \prec -predecessors in \mathcal{M} , then $m(\alpha)$ is undefined. It is, however, always defined if there is at least one such predecessor.

We are now ready to define consistency of constraints.

Definition 29. We call a constraint ψ consistent if for any element $\alpha \in \mathcal{E}(\psi)$ we have that $M(\alpha)$ is defined and that $m(\alpha) \prec M(\alpha)$ whenever $m(\alpha)$ is defined. A constraint that is not consistent is inconsistent.

So the consistency of the constraints only depends on the relationships between the constants from \mathcal{M} in ψ . This means

that any constraint network that does not contain any elements from \mathcal{M} is consistent. This is quite natural, as our constraints does not contain negation.

Lemma 30. *A constraint ψ has a solution if and only if it is consistent.*

Proof. Assume that ψ is inconsistent. Then there is an $\alpha \in \mathcal{E}(\psi)$ s.t. either $M(\alpha)$ is undefined, in which case we have two \prec -successors that do not overlap, or $m(\alpha) \not\prec M(\alpha)$. In the first case it should be obvious that there can be no solution. If the latter is true, we have two cases. If $\alpha \in \mathcal{M}(\psi)$ we have that $m(\alpha) = \alpha$, hence $\alpha \not\prec M(\alpha)$. This means that there is a set of \prec -successors $\beta_1, \dots, \beta_n \in \mathcal{M}(\psi)$ s.t. $\alpha \not\prec \bigotimes_i \beta_i$. Since this does not depend on a solution, this is always unsatisfiable.

If $\alpha \in \mathcal{GV}(\psi)$, then α must have at least one \prec -predecessor $\beta \in \mathcal{M}(\psi)$ and a set of \prec -successors $\beta_1, \dots, \beta_n \in \mathcal{M}(\psi)$ s.t. $\beta \not\prec \bigotimes_i \beta_i$, and we have the same situation as above.

It remains to prove that if ψ is consistent, there is a solution. However, such a solution can be found by setting $\sigma(\alpha) = M(\alpha)$ for each $\alpha \in \mathcal{GV}(\psi)$ which by definition is a solution. \square

It is easy to check consistency of a constraint, as it only amounts to computing the minimal limits and maximal elements, and then check the \prec -relationships between them.

However, consistency is not the only property we need for solving the constraints. It turns out that we can get hidden ambiguities through implicit disjunctions from the constants of \mathcal{M} , which is a problem for our construction of minimal models.

Definition 31. *We call a constraint ψ ambiguous if there is an element $\alpha \in \mathcal{GV}(\psi)$ and a set of elements $\beta_1, \dots, \beta_n \in \mathcal{M}(\psi)$ such that*

- $M(\alpha) \not\prec \beta_i$ for all $i \leq n$,
- *there is at most one i for which $\psi \models_{\mathcal{M}} O(\alpha, \beta_i)$,*
- *and where*

$$M(\alpha) \prec \bigoplus_{i=1}^n \beta_i$$

A constraint that is not ambiguous is unambiguous.

As we will see shortly, it is only possible to find a minimal model for unambiguous constraints. The intuition is that for an ambiguous constraint, there is an object that is contained in a sum of elements, but it is not clear how it should relate to each of the objects in the sum. The constraint will therefore not entail any relation between the element and the elements of the sum, but of course, there must be one. This is an instance of the general problem of obtaining minimal models of languages that allow disjunctions, e.g. disjunctive Datalog [4]. In our case, the disjunctions are hidden in the relationships between the constants of \mathcal{M} .

Example 32. *Assume we have*

$$\psi := A \prec \{0\} \wedge \{01\} \prec B_1 \wedge \{00\} \prec B_2$$

Then ψ is ambiguous. Any model must make A overlap at least one of the B_i , but none of the overlaps are entailed by ψ . Hence $\psi \not\models_{\mathcal{M}} O(A, B_1)$ and $\psi \not\models_{\mathcal{M}} O(A, B_2)$, although at least one of them must be the case in any model. Hence there can be no minimal model.

Adding $v \prec A \wedge v \prec B_1$ to ψ would still not make it unambiguous, as we now have both $\psi \not\models_{\mathcal{M}} A \prec B_1$ and $\psi \not\models_{\mathcal{M}} O(A, B_2)$. However, one of them must hold in any model.

Theorem 33. *If a consistent constraint has a minimal model, then it is unambiguous.*

Proof. We will prove the contrapositive, so assume ψ is ambiguous. Then there is an $\alpha \in \mathcal{GV}(\psi)$ s.t. there are some $\beta_1, \dots, \beta_n \in \mathcal{M}(\psi)$ where $M(\alpha) \not\prec \beta_i$ for each i , there is at most one β_j where $\psi \models_{\mathcal{M}} O(\alpha, \beta_j)$, and $M(\alpha) \prec \bigoplus_i \beta_i$.

For any model Q of ψ we must either have $Q \models \alpha \prec \beta_j$ or $Q \models O(\alpha, \beta_j) \wedge O(\alpha, \beta_{j'})$ for some $\beta_{j'} \neq \beta_j$. But neither of the two is entailed by ψ , hence Q cannot be minimal. Since Q was arbitrary, no such model can exist. \square

This means that constructing a minimal model is only meaningful for unambiguous constraints. However, one can always turn an ambiguous constraint to an unambiguous constraint by introducing some additional constraints settling the ambiguities. For an ambiguity over the element α , these additional constraint could either set α to be a part of one (or more) of the β s, or overlap at least one additional β .

This method could also be used to generate all possible solutions (with respect to the relationships between the elements), although there is an exponential number of such choices in the size of $\mathcal{GV}(\psi)$, so this would be unfeasible in the general case.

We will constructively prove the converse implication of Theorem 33 in Section VIII.

VII. CORRECT INDEX STRUCTURES

As stated earlier, a common use of bintrees, quad-trees, octrees and the like, is as spatial index structures. By construction, an index structure should be complete with respect to any spatial query. That is, a look-up should at least contain all the correct answers to the query. However, they are not always sound, they might contain incorrect answers. Therefore, a normal query procedure first makes a look-up in the index structure, and then decides using numerical algorithms which of the returned answers actually are correct.

In this section we will see how we can use the constraints introduced in the previous section to construct complete and sound index structures with respect to a set of mereological relations R . This will make querying more efficient as it will allow us to skip the refinement step, but more importantly, it will allow us to have queries involving spatial relations in a non-geospatial database.

Definition 34. *Assume that N is a geometrical model, $r_\varphi \in R \cup R_\delta$ a mereological relation, \vec{p} a tuple and that $\varphi \equiv$*

$\exists v_1 \dots \exists v_n \cdot \varphi'(\vec{p})$. Let the local completeness constraining function $\xi(\varphi(\vec{p}))$ be defined as

$$\xi(\exists v_1 \dots \exists v_n \cdot \varphi'(\vec{p})) = \varphi'(\vec{p})[v_1^{\varphi(\vec{p})}/v_1] \dots [v_n^{\varphi(\vec{p})}/v_n]$$

where each variable $v_i^{\varphi(\vec{p})}$ are unique for each formula φ and vector \vec{p} .

The idea is that any solution to the set of constraints returned by $\xi(\varphi)$ will make φ true. Hence, if we apply ξ to all true $(R \cup R_\delta)$ -statements in N , we will get a complete model that also works as an index structure.

Note that any mereological formula φ can be rewritten to a formula of the form $\exists v_1 \dots \exists v_n \cdot \varphi'(\vec{p})$, so the assumption made in the definition does not restrict the number of formulas ξ can be applied to.

The construction of the constraints from R_δ is almost the same procedure as when one constructs a bintree as a spatial index structure, with the only difference being that we do not set the representation of an element α to be equal to all its overlapping blocks at depth δ . We rather construct upper and lower bounds of α by using the relations $\beta \prec \alpha$ and $\alpha \prec \beta$, and then set it to overlap all the blocks it overlaps at depth δ by using $O(\beta, \alpha)$.

Definition 35. Let the global completeness constraining function Ξ_δ , for some initial depth δ , be defined as

$$\Xi_\delta^R(N) = \bigwedge_{r_\varphi \in R \cup R_\delta} \bigwedge_{N \models_{\mathcal{N}} \varphi(\vec{p})} \xi(\varphi(\vec{p}))$$

Note that many of the constraints generated by the relations from R_δ are redundant. For instance if we have $\alpha \prec \beta$ for some $\alpha \in \mathcal{G}$ and $\beta \in \mathcal{M}$, we could also have $\alpha \prec \beta'$ for some β' where $\beta \prec \beta'$. For simplicity, we will assume that we only keep the constraints $\alpha \prec \beta_M$ for the smallest β_M , and $\beta_m \prec \alpha$ for the largest β_m .

Lemma 36. $Q \models \xi(\varphi(\vec{p}))\sigma \Leftrightarrow Q \models \varphi(\vec{p})$ for any mereological model Q , mereological formula φ and solution σ .

Proof. Assume $\varphi(\vec{p}) = \exists v_1 \dots \exists v_n \cdot \varphi'(\vec{p})$. Then

$$\begin{aligned} Q \models \xi(\varphi(\vec{p}))\sigma &\Leftrightarrow Q \models \left(\varphi'(\vec{p}) \left[v_1^{\varphi(\vec{p})}/v_1 \right] \dots \left[v_n^{\varphi(\vec{p})}/v_n \right] \right) \sigma \\ &\Leftrightarrow Q \models \varphi'(\vec{p}) \left[\sigma \left(v_1^{\varphi(\vec{p})} \right) / v_1 \right] \dots \left[\sigma \left(v_n^{\varphi(\vec{p})} \right) / v_n \right] \\ &\Leftrightarrow Q \models \varphi'(\vec{p})[a_1/v_1] \dots [a_n/v_n] \\ &\Leftrightarrow Q \models \exists v_1 \dots \exists v_n \cdot \varphi'(\vec{p}) \\ &\Leftrightarrow Q \models \varphi(\vec{p}) \end{aligned}$$

for $\sigma \left(v_i^{\varphi(\vec{p})} \right) = a_i$. \square

Now that we have constraints properly describing the geometries, we want to construct a minimal model of these constraints. This model will then only entail what the constraints entail, which is exactly the true sentences in the model N . Hence, we have a sound and complete model that also functions as a spatial index at the initial depth δ .

According to the definition of Ξ_δ^R , we need to know all true statements of N with respect to the relations of $R \cup R_\delta$.

This would amount to computing all possible relationships between all possible elements of \mathcal{G} . However, if we start by computing the constraints with respect to the index relations R_δ , we can use $M(\alpha)$ with respect to these constraints as a normal bintree index structure for α . This index structure can be used to determine which objects might be related with a given relation in the same way as a normal spatial index.

However, before we can attempt to solve our constraints, we need to know that they are consistent and unambiguous. If the constraints are constructed from true sentences in a model N , they must be consistent. The following lemma states the unambiguity.

Lemma 37. $\Xi_\delta^R(N)$ is unambiguous.

Proof. Since the relations of R_δ determines the relationship between every pair of $\alpha \in \mathcal{GV}(\Xi_\delta^R(N))$ and $\beta \in \mathcal{M}_\delta$ in $\Xi_\delta^R(N)$, there can be no ambiguity in the constraints. \square

Theorem 38. We have

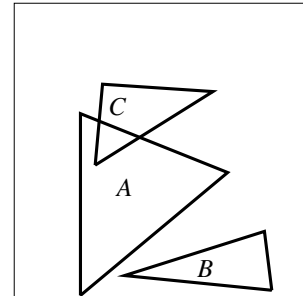
$$\Xi_\delta^R(N) \models_{\mathcal{M}} \varphi(\vec{p}) \Leftrightarrow N \models \varphi(\vec{p})$$

for any $r_\varphi \in R \cup R_\delta$ and $\vec{p} \in \mathcal{G}^*$.

Proof. (\Rightarrow): Assume $\Xi_\delta^R(N) \models_{\mathcal{M}} \varphi(\vec{p})$, $\Xi_\delta^R(N) = \bigwedge_i \xi(\varphi_i)$ and that Q is a model of $\Xi_\delta^R(N)$. We know, by lemma 36, that $Q \models \xi(\varphi)\sigma \Leftrightarrow Q \models \varphi$. This means that $Q \models \Xi_\delta^R(N)\sigma \Leftrightarrow Q \models \bigwedge_i \varphi_i$. Hence $Q \models \bigwedge_i \varphi_i \Rightarrow Q \models \varphi(\vec{p})$ for any model Q , so $\bigwedge_i \varphi_i \models_{\mathcal{M}} \varphi(\vec{p})$. By Theorem 15 we get $\bigwedge_i \varphi_i \models_{\mathcal{N}} \varphi(\vec{p})$. Since $N \models \bigwedge_i \varphi_i$, we get $N \models \varphi(\vec{p})$.

(\Leftarrow): This follows easily from Lemma 36. \square

Example 39. Let's construct a toy example with three two-dimensional areas, and assume that $\mathcal{G} = \{A, B, C\}$ and that the following is a visualisation of a geometrical model N :



We will assume that $R = \{O\}$. For simplicity let the initial depth δ of the bintree be 4 and that we start dividing along the y-axis. If we draw the blocks at the depth 4 over the geometries, we get

0000	0010	1000	1010
0001	0011	1001	1011
0100	0101	1100	1110
0101	0111	1101	1111

If we apply $\Xi_4^{\{O\}}$ to the model N we get

$$\begin{aligned}
\Xi_4^{\{O\}}(N) = & \\
& A \prec \{01, 0001, 0011, 1001, 110\} \wedge \{0101\} \prec A \wedge \\
& v_1 \prec A \wedge v_1 \prec \{0001\} \wedge v_2 \prec A \wedge v_2 \prec \{0011\} \wedge \\
& v_3 \prec A \wedge v_3 \prec \{1001\} \wedge v_4 \prec A \wedge v_4 \prec \{0100\} \wedge \\
& v_5 \prec A \wedge v_5 \prec \{1100\} \wedge v_6 \prec A \wedge v_6 \prec \{0101\} \wedge \\
& v_7 \prec A \wedge v_7 \prec \{0101\} \wedge v_8 \prec A \wedge v_8 \prec \{0111\} \wedge \\
& v_9 \prec A \wedge v_9 \prec \{1101\} \wedge B \prec \{0111, 1101, 1111\} \wedge \\
& z_1 \prec B \wedge z_1 \prec \{0111\} \wedge z_2 \prec B \wedge z_2 \prec \{1101\} \wedge \\
& z_3 \prec B \wedge z_3 \prec \{1111\} \wedge C \prec \{0011, 1001, 0101\} \wedge \\
& x_1 \prec C \wedge x_1 \prec \{0011\} \wedge x_2 \prec C \wedge x_2 \prec \{1001\} \wedge \\
& x_3 \prec C \wedge x_3 \prec \{0101\} \wedge y \prec C \wedge y \prec A
\end{aligned}$$

The constraints constructed in this section constrain all elements of \mathcal{G} at once, so once we have a solution to these constraints, the entire procedure can be viewed as a bulk load insertion. However, a single insert procedure for an element p can easily be derived from the procedure above. Assume we have a model Q with correct representations of the elements of \mathcal{G} according to the geometrical model N . Assume we want to insert p . First, note that to determine the truth of $\varphi(\vec{p})$ in N we only need to consult N when p is an element of \vec{p} . Everything else can be decided by querying Q . We would then construct $\psi_\delta := \bigwedge_{r_\varphi \in R_\delta, N \models \varphi(p)} \xi(\varphi(p))$, compute $M(p)$ with respect to ψ_δ , and let $R_p = \{p\} \cup \{p' \in \mathcal{G} \mid Q \models O(p', M(p))\}$. The constraints we would need to solve to construct correct representations of $\mathcal{G} \cup \{p\}$ would then be given by

$$\begin{aligned}
\psi := & \psi_\delta \wedge \bigwedge_{r_\varphi \in R \cup R_\delta} \left(\bigwedge_{\vec{p} \in \mathcal{G} \setminus R_p, Q \models \varphi(\vec{p})} \xi(\varphi(\vec{p})) \right) \\
& \wedge \bigwedge_{r_\varphi \in R} \left(\bigwedge_{\vec{p} \in \mathcal{G} \cap R_p, p \in \vec{p}, N \models \varphi(\vec{p})} \xi(\varphi(\vec{p})) \right)
\end{aligned}$$

Note that the only representations of the elements of \mathcal{G} that are affected by the insert of p , must overlap $M(p)$ with respect to ψ_δ .

VIII. SOLVING THE CONSTRAINTS

Now that we know which models we are interested in, we can define an algorithm for solving the constraints. In this section we will construct a solver that solves the constraints in polynomial time in size of the number of conjuncts in

the constraints. The space consumption of the returned representations are, however, far from optimal, and a solver that returns optimal representation is left as future work. The solver presented in this section is therefore mostly to prove that the problem of finding a minimal solution to a constraint is in the complexity class PTIME in the size of the constraint graph.

In this section, we will assume that ψ (as a graph) is without cycles. As cycles only would lead to equal elements, they can easily be removed under the constraint solving process by setting in one element that represents all elements in the cycle. We can then reintroduce them when we have a solution by setting the solution of each element in the cycle to equal that of the representing element.

We will also extend our operators and relations to be defined for \emptyset , such that we do not always have to check whether results are empty. We let for any $\alpha \in \mathcal{M}$:

- $\alpha \neq \emptyset$,
- $\emptyset \prec \alpha$,
- $\emptyset \oplus \alpha = \alpha$ and $\alpha \oplus \emptyset = \alpha$,
- $\emptyset \ominus \alpha = \emptyset$ and $\alpha \ominus \emptyset = \alpha$,
- $\emptyset \otimes \alpha = \emptyset$ and $\alpha \otimes \emptyset = \emptyset$.

This makes (\mathcal{M}, \prec) a lattice. Note that this is a purely syntactic extension, and is not part of the semantics. We say that an element is undefined if it is equal to \emptyset . Note that we now always get a value from $m(\alpha)$ and $M(\alpha)$.

For our solver, we will need a syntactic way of finding all $\beta \in \mathcal{M}(\psi)$ such that $\psi \models_{\mathcal{M}} O(\alpha, \beta)$ for each $\alpha \in \mathcal{GV}(\psi)$. This can be done by finding all constants $\beta \in \mathcal{M}(\psi)$ such that either

- $O(m(\alpha), \beta)$,
- $M(\alpha) \prec \beta$,
- or there is an element $v \in R_\prec^\alpha \cap R_\prec^\beta$.

Using this, we will also be able to syntactically compute the following necessary function.

Definition 40. Assume ψ is a constraint. Let

$$B_{-O}^\alpha := \bigoplus_{\beta \in \mathcal{M}(\psi), \psi \not\models_{\mathcal{M}} O(\alpha, \beta)} \beta$$

and define

$$M'(\alpha) := M(\alpha) \ominus B_{-O}^\alpha$$

Definition 41. Let $\varpi_\psi : \mathcal{GV}(\psi) \rightarrow \mathcal{B}$ be a function returning a unique block of length $\lceil \log_2 |\mathcal{GV}(\psi)| \rceil$ for each $\alpha \in \mathcal{GV}(\psi)$.

Algorithm 1 Function that finds the minimal solution to ψ .

```

function solve( $\psi$ )
   $\delta := \max_{\beta \in \mathcal{M}(\psi)} \Delta(\beta)$ 
  for  $\alpha \in \mathcal{GV}(\psi)$  do
     $\sigma_0(\alpha) := \{s \circ 0 \circ \varpi_\psi(\alpha) \mid \{s\} \prec M'(\alpha), |s| = \delta\}$ 
  for  $\alpha \in \mathcal{GV}(\psi)$  do
     $\sigma_\mu(\alpha) := m(\alpha) \oplus \bigoplus_{\alpha' \in R_\prec^\psi(\alpha) \cap \mathcal{GV}(\psi)} \sigma_0(\alpha')$ 
  return  $\sigma_\mu$ 

```

Definition 42. Assume ψ to be an unambiguous, consistent constraint. Let $\mu(\psi)$ be the model induced by $\text{solve}(\psi)$ in algorithm 1.

The main idea behind the function solve is to first construct an initial substitution σ_0 that entails as few relationships between the objects of ψ as possible, and then propagate the necessary parts upwards to construct the correct solution σ_μ .

The first for-loop constructs the initial substitution σ_0 . Since ϖ_ψ returns unique blocks of equal lengths we have for any $\gamma, \gamma' \in \mathcal{GV}(\psi)$, that $\sigma_0(\gamma)$ and $\sigma_0(\gamma')$ are disjoint. Furthermore, for any $\alpha \in \mathcal{GV}(\psi)$ and any $\beta, \beta' \in \mathcal{M}(\psi)$, we have both $\sigma_0(\alpha) \prec \beta$ if and only if $\psi \models_{\mathcal{M}} \alpha \prec \beta$ and $\beta' \not\prec \sigma_0(\alpha)$ (for details, see the proof of lemma 44). However, $\sigma_0(\alpha) \prec M(\alpha)$ by construction.

So σ_0 constructs representations that entail as few sentences as possible, and only sentences already entailed by the constraints. The second for-loop can now iterate the elements of $\mathcal{GV}(\psi)$ and constructs the correct solution such that every element contains the elements the constraints force them to contain. This step is somewhat similar to a more traditional chase algorithm [4], although instead of adding triples to a relation, we add blocks to representations.

The next example and the following lemmas and their accompanying proofs will give the reader a more detailed insight into the correctness of the algorithm.

Example 43. Assume that $\mathcal{G} = \{A, B, C\}$,

$$\begin{aligned} \psi &:= A \prec \{0\} \wedge \{01\} \prec B \wedge v \prec A \\ &\quad v \prec C \wedge C \prec B \wedge C \prec \{00, 10\} \end{aligned}$$

and that

$$\begin{aligned} \varpi_\psi(A) &= 00 & \varpi_\psi(B) &= 01 \\ \varpi_\psi(C) &= 10 & \varpi_\psi(v) &= 11 \end{aligned}$$

(since $\lceil \log_2 |\mathcal{GV}(\psi)| \rceil = \lceil \log_2 4 \rceil = 2$). We will now go through each step of the computation of $\mu(\psi)$:

(i) First, $\delta = \max_{\beta \in \mathcal{M}(\psi)} \Delta(\beta) = 2$.

(ii) We continue by computing $M(A) = \{0\}$ and $B_{-O}^A = \{01\}$. Now $M'(A) = M(A) \ominus B_{-O}^A = \{00\}$. Hence $\sigma_0(A) = \{00 \circ 0 \circ 00\} = \{00000\}$.

Computing the same for B, C and v , we get $\sigma_0(B) = \{00001, 01001, 10001, 11001\}$, $\sigma_0(C) = \{00010, 10010\}$ and $\sigma_0(v) = \{00011\}$.

(iii) We can now compute σ_μ . So

$$\begin{aligned} \sigma_\mu(A) &= m(A) \oplus \sigma_0(A) \oplus \sigma_0(v) \\ &= \emptyset \oplus \{00000\} \oplus \{00011\} \\ &= \{00000, 00011\} \end{aligned}$$

Doing the same for B, C and v , we get $\sigma_\mu(B) = \{01, 0001, 00001, 10001, 10010, 11001\}$, $\sigma_\mu(C) = \{0001, 10010\}$ and $\sigma_\mu(v) = \{00011\}$.

We can now see that e.g. $O(\sigma_\mu(A), \sigma_\mu(C))$ and that $\sigma_\mu(C) \prec \sigma_\mu(B)$, but $\neg O(\sigma_\mu(A), \{01\})$.

Lemma 44. Assume ψ is a consistent, unambiguous constraint. We have for any $\alpha, \beta \in \mathcal{E}(\psi)$ that

$$\psi \models_{\mathcal{M}} \alpha \prec \beta \Leftrightarrow \sigma_\mu(\alpha) \prec \sigma_\mu(\beta)$$

where σ_μ results from $\text{solve}(\psi)$.

Proof. (\Rightarrow): This is easy to see from the construction of σ_μ .

(\Leftarrow): We will prove the contrapositive through proof by contradiction. So assume $\psi \not\models_{\mathcal{M}} \alpha \prec \beta$, but $\sigma_\mu(\alpha) \prec \sigma_\mu(\beta)$. We have that

$$\begin{aligned} \sigma_\mu(\alpha) &= m(\alpha) \oplus \sigma_0(\alpha) \oplus \sigma_0(\alpha_1) \oplus \dots \oplus \sigma_0(\alpha_n) \\ \sigma_\mu(\beta) &= m(\beta) \oplus \sigma_0(\beta) \oplus \sigma_0(\beta_1) \oplus \dots \oplus \sigma_0(\beta_m) \end{aligned}$$

for $\alpha_1, \dots, \alpha_n$ where $\psi \models_{\mathcal{M}} \alpha_i \prec \alpha$ for each α_i , and β_1, \dots, β_m where $\psi \models_{\mathcal{M}} \beta_i \prec \beta$. It must therefore be the case that $\sigma_0(\alpha) \prec m(\beta) \oplus \sigma_0(\beta) \oplus \sigma_0(\beta_1) \oplus \dots \oplus \sigma_0(\beta_m)$. Since $\psi \not\models_{\mathcal{M}} \alpha \prec \beta$ there is no $\beta_i = \alpha$. We now have three cases: Either $\alpha \in \mathcal{GV}(\psi)$ and $\beta \in \mathcal{M}(\psi)$; $\alpha \in \mathcal{M}(\psi)$ and $\beta \in \mathcal{GV}(\psi)$; or lastly, both $\alpha, \beta \in \mathcal{GV}(\psi)$.

In the first case, we have that $\sigma_\mu(\beta) = \beta$, because for every β_i , we have that $\sigma_0(\beta_i) \prec M(\beta_i) \prec \beta$. So $\sigma_0(\alpha) \prec \beta$. However, since $\sigma_0(\alpha)$ contains only blocks that have a prefix among the blocks of $M(\alpha) \ominus B_{-O}^\alpha$, it must be the case that $M(\alpha) \ominus B_{-O}^\alpha \prec \beta$. This further implies $M(\alpha) \prec \beta \oplus B_{-O}^\alpha$. Since $M(\alpha) \prec \beta$ implies $\psi \models_{\mathcal{M}} \alpha \prec \beta$ we must have $M(\alpha) \not\prec \beta$. Furthermore, $\psi \models_{\mathcal{M}} O(\alpha, \beta)$, since if not then $\beta \prec B_{-O}^\alpha$ and $\sigma_0(\alpha) \prec M(\alpha) \ominus B_{-O}^\alpha$. However, $M(\alpha) \prec \beta \oplus B_{-O}^\alpha$, $M(\alpha) \not\prec \beta$ and $\psi \models_{\mathcal{M}} O(\alpha, \beta)$ implies that ψ is ambiguous, which is a contradiction.

In the second case, we have that $\sigma_\mu(\alpha) = \alpha$. We cannot have $\alpha \prec m(\beta)$, since this would imply $\psi \models_{\mathcal{M}} \alpha \prec \beta$. Hence, $\alpha \prec \sigma_0(\beta) \oplus \sigma_0(\beta_1) \oplus \dots \oplus \sigma_0(\beta_m)$. Since $\sigma_0(\beta)$ and each $\sigma_0(\beta_i)$ are all constructed of blocks at depth that of $\delta + 1 + \lceil \log_2 |\mathcal{GV}(\psi)| \rceil$, they can therefore not sum up to an element containing any block with depth less than that of $1 + \max_{\beta \in \mathcal{M}(\psi)} \Delta(\beta)$. Hence, α cannot be part of such a sum, so we have arrived at a contradiction.

In the third case, we know that for any $\gamma, \gamma' \in \mathcal{GV}(\psi)$, by the uniqueness and length of $\varpi_\psi(\gamma)$ and $\varpi_\psi(\gamma')$, we have that $\sigma_0(\gamma), \sigma_0(\gamma')$ are disjoint. So for $\sigma_0(\alpha) \prec m(\beta) \oplus \sigma_0(\beta) \oplus \dots \oplus \sigma_0(\beta_m)$ to hold, we must either have $\sigma_0(\alpha) \prec m(\beta)$, or that there is a β_i where $\alpha = \beta_i$ (by $=$ we mean that they denote the same element in the constraints). For the first case, we can argue similarly as in the first case of the proof and arrive at a contradiction, and in the second case we would have $\psi \models_{\mathcal{M}} \alpha \prec \beta_i$, which implies $\psi \models_{\mathcal{M}} \alpha \prec \beta$, which also is a contradiction. \square

Lemma 45. Assume ψ is a consistent, unambiguous constraint. We have for any $\alpha_1, \dots, \alpha_n \in \mathcal{E}(\psi)$ that

$$\begin{aligned} \psi \models_{\mathcal{M}} \exists v (v \prec \alpha_1 \wedge \dots \wedge v \prec \alpha_n) &\Leftrightarrow \\ \sigma_\mu(\alpha_1) \otimes \dots \otimes \sigma_\mu(\alpha_n) &\in \mathcal{M} \end{aligned}$$

where σ_μ results from $\text{solve}(\psi)$.

Proof. For $\exists v (v \prec \alpha_1 \wedge \dots \wedge v \prec \alpha_n)$ to hold in all models of ψ , all of the α_i s must always share some common

part. This common part must either be a constant, in case of overlapping lower limits, or a variable explicitly set to be a predecessor of all α_i in ψ^* . So $\psi \models_{\mathcal{M}} \exists v (v \prec \alpha_1 \wedge \dots \wedge v \prec \alpha_n)$ holds iff either $\bigotimes_{i=1}^n m(\alpha_i) \in \mathcal{M}$ or $\exists v \in \mathcal{GV}(\psi) ((\bigwedge_{i=1}^n v \prec \alpha_i) \in \psi^*)$. All other cases can be reduced to one of the two.

Since all $\sigma_0(\alpha)$ are disjoint, for $\bigotimes_{i=1}^n \sigma_\mu(\alpha) \in \mathcal{M}$ to hold, we must have that either $\bigotimes_{i=1}^n m(\alpha_i) \in \mathcal{M}$, or that there is some $\alpha \in \mathcal{GV}(\psi)$ s.t. $\sigma_0(\alpha)$ is a part in each of the sums $\sigma_\mu(\alpha_i)$. However, the last case holds iff $(\bigwedge_{i=1}^n \alpha \prec \alpha_i) \in \psi^*$. \square

Theorem 46. *Assume ψ is a consistent, unambiguous constraint. Then $\mu(\psi)$ is a minimal model.*

Proof. By construction of σ_μ it must be a solution to ψ when ψ is consistent, that is $\psi \sigma_\mu$ is valid. Furthermore, since $\mu(\psi)$ is induced by a solution to ψ , it is a model of ψ . Since $\mu(\psi)$ is a model of ψ for consistent ψ , by definition of entailment of a constraint, it must be the case that $\psi \models_{\mathcal{M}} \varphi \Rightarrow \mu(\psi) \models_{\mathcal{M}} \varphi$.

It remains to prove $\mu(\psi) \models_{\mathcal{M}} \varphi \Rightarrow \psi \models_{\mathcal{M}} \varphi$. Without loss of generality we can assume that $\varphi \equiv \bigwedge_i \alpha'_i \prec \beta'_i \wedge \exists \vec{v}. \bigwedge_j \alpha_j \prec \beta_j$, where $\alpha'_i, \beta'_i \in \mathcal{G}(\psi) \cup \mathcal{M}(\psi)$ for each i , and $\alpha_j, \beta_j \in \mathcal{G}(\psi) \cup \mathcal{M}(\psi) \cup \mathcal{V}$ for each j . By Lemma 44 we have that $\psi \models_{\mathcal{M}} \alpha \prec \beta \Leftrightarrow \mu(\psi) \models_{\mathcal{M}} \alpha \prec \beta$, for $\alpha, \beta \in \mathcal{G}(\psi) \cup \mathcal{M}(\psi)$, so it remains to prove the result for $\varphi' \equiv \exists \vec{v}. \bigwedge_j \alpha_j \prec \beta_j$.

We will prove this contrapositively, so assume $\psi \not\models_{\mathcal{M}} \varphi'$. Then there must be at least one model Q where $Q \models \neg \exists \vec{v}. \varphi'$. We can compute the upper and lower bounds of each variable v_i from \vec{v} in φ' , $M(v_i)$ and $m(v_i)$ resp., in the same manner as for the constraints. We then have $M(v_i) = \bigotimes_i c_i \otimes \bigotimes_j \alpha_j$, where $c_i \in \mathcal{M}(\varphi'), \alpha_j \in \mathcal{G}(\varphi')$ are the constants set greater than v_i in $(\varphi')^*$. Since $Q \models \neg \exists \vec{v}. \varphi'$ it must be the case that either there is some $v_i \in \vec{v}$ s.t. $M(v_i)$ is undefined, or $m(v_i)$ is defined for at least one v_i (if not then $M(v_i)$ would be valid solution of v_i) and that $Q \models m(v_i) \not\prec M(v_i)$.

In the first of the two cases, we have $\psi \not\models_{\mathcal{M}} \exists v (\bigwedge_i v \prec \alpha_i)$ where the α_i s are successors of v_i in φ' . By Lemma 45 we then have that $\bigotimes_i \sigma_\mu(\alpha_i) \notin \mathcal{M}$, so $\mu(\psi) \not\models_{\mathcal{M}} \exists \vec{v}. \varphi'$.

In the second of the two cases, we must have $m(v_i) = \bigoplus_i k_i \oplus \bigoplus_j \beta_j$ where $k_i \in \mathcal{M}(\varphi'), \beta_j \in \mathcal{G}(\varphi')$ are the elements set to be part of v_i in $(\varphi')^*$. For $Q \models m(v_i) \not\prec M(v_i)$ to be the case, there must be at least one pair of elements $\gamma, \gamma' \in \mathcal{G}(\varphi') \cup \mathcal{M}(\varphi')$ such that $\gamma = k_i$ or $\gamma = \beta_j$ and $\gamma' = c_i$ or $\gamma' = \alpha_j$, but where $Q \models \gamma \not\prec \gamma'$. However, this implies $\psi \not\models_{\mathcal{M}} \gamma \prec \gamma'$. Since $\gamma, \gamma' \in \mathcal{G}(\psi) \cup \mathcal{M}(\psi)$, we have by Lemma 44 that $\mu(\psi) \not\models \gamma \prec \gamma'$, and furthermore, $\mu(\psi) \not\models \exists \vec{v}. \varphi'$. \square

IX. IMPLEMENTATION IN RELATIONAL ALGEBRA AND OTHER QUERY LANGUAGES

Now we have seen when it is possible to solve mereological constraints, and how one can construct such solutions. However, apart from proving the existence of sums, products and differences of our representations, an algorithm for

constructing representations from constraints, and a proof of correctness, we still have not addressed how the mereological relations actually can be evaluated over a relational database containing these representations. In this section these details will be outlined.

We will assume that a mereological model Q is implemented as a relation Q s.t. $Q(s, a)$ iff $s \in a^Q$. Representing our bintree blocks as bit-strings was natural for theoretical treatment. However, in this section we will assume them to be integers, as this is a more natural representation for actual implementation. We will still have that a block s_1 is part of a block s_2 if s_2 's bit-representation is a prefix of s_1 's bit-representation.

We will start by defining the \prec -relation in a more procedural manner adopting this new representation of our blocks:

$$s \prec s' := (s' = (s \gg |s| - |s'|))$$

where $|s| := 1 + \lfloor \log_2 s \rfloor$ is the length of the bit-representation of the integer s , and \gg is right bit-shift.

We can now use \prec to compute \prec over elements of Q as

$$\begin{aligned} \prec &:= (\pi_2(Q) \times \pi_2(Q)) - \\ &\pi_{1,3}((Q \times \pi_2(Q)) - \pi_{1,2,4}(\sigma_{1 \prec 3}(Q \times Q))) \end{aligned}$$

where $\pi_F, \sigma_\varphi, \times$ are all from standard relational algebra (see e.g. [4]), and is projection, selection and cross-product of the tuples in relations, respectively. Note that $\prec := \pi_{1,3}((Q \times \pi_2(Q)) - \pi_{1,2,4}(\sigma_{1 \prec 3}(Q \times Q)))$.

If we want to compute a window query, that is, a query with a constant $\beta \in \mathcal{M}$, then we have

$$\begin{aligned} X \prec \beta &:= \pi_2(Q) - \pi_2(Q - \pi_{1,2}(\sigma_{1 \prec 3}(Q \times \beta))) \\ \beta \prec X &:= \pi_2(Q) - \pi_2(Q - \pi_{2,3}(\sigma_{1 \prec 2}(\beta \times Q))) \end{aligned}$$

We could easily translate a numerically represented geometry to an element of \mathcal{M}_δ by using a standard bintree construction, and then use this element in a window query. Since all elements of Q are correctly represented according to R_δ our system would return the correct answers with respect to a resolution of δ .

There is also another suitable representation of our blocks, which allows us to get rid of the computation of the logarithm in \prec . This representation stores the depth of each bit-string along with the bit-string, such that each block is a pair (l, s) where l is the length and s is the bit-string. This second representation allows for the simpler definition of \prec :

$$(l, s) \prec (l', s') := (s' = (s \gg l - l'))$$

We would then have Q as a relation of arity 3, such that $Q(l, s, a)$, and must then update the projections in the definition of \prec accordingly.

Intersection \otimes , union \oplus , and complement of bintrees is implemented and discussed in [6]. Their implementations all have a linear complexity in the size of the blocks in their arguments. We use intersection and complement to define difference in the standard way, $a \ominus b := a \otimes b^{-1}$.

To compute our mereological relations over Q , we have to get rid of the existentially quantified variables, as they actually do not denote an object in the database but rather an object of \mathcal{M} . To do this we can just substitute each variable v with the intersection of constants and free variables denoting its maximum bound $M(v)$. The entire query will then look like $\bigwedge_i \alpha_i \prec \bigotimes_j \beta_{i,j}$, where all α_i are elements of $\mathcal{G} \cup \mathcal{M} \cup \mathcal{V}$ and free variables that ranges over \mathcal{G} , and $\beta_{i,j}$ are elements of $\mathcal{M} \cup \mathcal{G}$ or free variables over elements of \mathcal{G} . Whenever $\alpha_i \in \mathcal{V}$, we can rewrite $\alpha_i \prec \bigotimes_j \beta_{i,j}$ to $\text{EXISTS}(\bigotimes_j \beta_{i,j})$, where EXISTS test whether its argument is empty (and is a standard keyword in SQL). If we assume that I contains all indices i where $\alpha_i \in \mathcal{V}$ and I' the rest, the entire query can be rewritten to

$$\bigwedge_{i \in I} \text{EXISTS}(\bigotimes_j \beta_{i,j}) \wedge \bigwedge_{i \in I'} \bigwedge_j \alpha_i \prec \beta_{i,j}$$

In Datalog with negation, assuming we have the implementation of \prec as above (either in an arithmetic extension of Datalog or as an external predicate) \prec is defined by the following rule:

$$\prec(X, Y) \leftarrow \text{not}(Q(S, X), \text{not}(Q(S', Y), \prec(S, S'))).$$

assuming $Q(s, a)$ iff $s \in a^Q$, as above.

In SPARQL, \prec could be implemented as `:partOf` as

```
CONSTRUCT { ?a :partOf ?b . }
WHERE
{
  ?a a :Geo .
  ?b a :Geo .
  FILTER NOT EXISTS
  {
    ?s :Q ?a .
    FILTER NOT EXISTS
    {
      ?s2 :Q ?b .
      FILTER (?s <= ?s2)
    }
  }
}
```

assuming \prec is implemented as \leq , and $s :Q p$ iff $s \in p^Q$.

X. COMPLEXITY

We will now turn to the actual complexity of computing our minimal model $\mu(\psi)$ and the space complexity of the final representations.

Theorem 47. *Assume ψ is a consistent, unambiguous constraint with. Let $n = |\mathcal{E}(\psi)|$, m be the number of conjuncts in ψ and k be the largest cardinality of any element of $\mathcal{M}(\psi)$. The time complexity of computing $\mu(\psi)$ is $\mathcal{O}(m^3 + n^2k)$.*

Proof. We have that the algorithmic complexity of computing

- $a \oplus b$, $a \otimes b$ and $a \ominus b$ are all $\mathcal{O}(k)$ [6],
- $a \prec b$ is $\mathcal{O}(k)$ (can be reduced to checking $a \otimes b = a$),
- the transitive closure of a graph is $\mathcal{O}(m^3)$ [7],
- $M(\alpha)$ is $\mathcal{O}(n^2k)$ for each α ,
- $m(\alpha)$ is $\mathcal{O}(n^2k)$ for each α ,
- ψ^* is $\mathcal{O}(m^3 + n^2k)$,

- the set of elements that are forced to overlap α (given ψ^*) is $\mathcal{O}(n^2k)$ for each α ,
- B_{-O}^α (given ψ^*) is $\mathcal{O}(n^2k)$ for each α ,
- $M'(\alpha)$ (given ψ^*) is $\mathcal{O}(n^2k)$ for each α .

This means that the complexity of computing each of the for-loops in the algorithm, assuming that we already have computed ψ^* , and B_{-O}^α , $M(\alpha)$ and $m(\alpha)$ for each α , is $\mathcal{O}(nk)$, and $\mathcal{O}(nk)$, giving a combined complexity of $\mathcal{O}(m^3 + n^2k)$. \square

Lemma 48. *Assume ψ is a consistent unambiguous constraint with a fixed maximum depth δ . Let $n = |\mathcal{GV}(\psi) \cup \mathcal{M}(\psi)|$ and $m = |\mathcal{G}(\psi)|$. We have that the storage space required by the representations returned from $\mu(\psi)$ is bound by $\mathcal{O}(mn \log n)$.*

Proof. Every $\sigma_0(\alpha)$ will after the first for-loop use $2^{\delta-1}(\delta + 1 + \lceil \log_2 n \rceil)$ bits of storage. $m(\alpha)$ can take up $n2^{\delta-1}$ space, which means that each $\sigma_\mu(\alpha)$ can, worst case, use $n2^{\delta-1} + n2^{\delta-1}(\delta + 1 + \lceil \log_2 n \rceil)$ bits. This means that the entire model of the m elements of \mathcal{G} , $\mu(\psi)$, has a space consumption bound by $\mathcal{O}(mn \log n)$ (for a fixed δ). \square

The depth only decides the resolution of the constraining constants, so we can easily set a maximum depth for most applications.

Note that the space needed to store the representations from solve depends on the size of the graph. The size of the graph depends on the number of witnesses variables we introduce, so the size of our representations will depend on the number of tuples in the relations, just like the naive solution. In the next section, we will outline a solution to this, which we are currently working on.

XI. CONCLUSION AND FUTURE WORK

We have seen that we can in polynomial time construct sound and complete index structures. These structures allow us to pose mereological queries over objects over a normal relational database.

Our first priority is to find a solution returning optimal representations. We are currently working on a solver using the transitive closure compression scheme from [8]. This algorithm assigns a number and a set of intervals to each node in directed acyclic graphs. The intervals of each node contains the numbers of this node's reachable nodes. In the paper, they also describe how one can obtain optimal compression schemes. Our idea is to use this optimal compression scheme and assign optimal representations from \mathcal{M} to the intervals in the compression scheme. We think this would give representations of size $\mathcal{O}(n^2 \log n)$ where $n = |\mathcal{G}|$. Another potential optimisation is based on the observation that we do not really need to construct $\sigma_0(\alpha)$ for all $\alpha \in \mathcal{GV}(\psi)$. In fact, it seems that we only need to construct σ_0 for \prec -minimal elements in the graph, and elements $\alpha \in \mathcal{GV}(\psi)$ that has exactly the same \prec -predecessors as another element in $\mathcal{GV}(\psi)$. Furthermore, there might be many redundant variables and edges in the constraint graph that we can remove, e.g. all variables $v \in \mathcal{V}(\psi)$ where there exists an element $\alpha \in \mathcal{GV}(\psi)$

such that $R_{\prec}^{\psi}(v) \subseteq R_{\prec}^{\psi}(\alpha)$ and $R_{\succ}^{\psi}(v) \subseteq R_{\succ}^{\psi}(\alpha)$ is redundant and can be removed from the constraints.

In the future, we also plan to make an implementation of the system, such that we can test the actual performance over real data. We also want to extend the system to include a *touching*-relation and a projection function. The first of the two will allow us to express mereotopological relations and constraints. With such a system one could formalise interesting calculi like RCC8 [5] and Allen's Interval Algebra [9]. There has also been done work on touching relations on quad-trees [10], which should be easy to generalise to bintrees.

With a projection function, we can represent geometries that change shape, size and location over time. Such a function is very easy to implement, as projecting a block down one dimension only involves removing the i 'th bit in each n -bit sequence of the block. It is not trivial, however, to construct a solution of a constraint that constrains objects in different dimensions.

If we combine the two extensions, we can construct correct mereotopological representations of spatio-temporal geometries, and the much more expressive corresponding base relations.

Another interesting research topic is whether it is possible to extend the expressiveness of our query language beyond conjunctive queries to other first order query languages, without losing feasibility of solving the constraints.

REFERENCES

- [1] M. Koubarakis, *Spatio-temporal databases: The CHOROCHRONOS approach*. Springer Science & Business Media, 2003, vol. 2520. [Online]. Available: <http://dx.doi.org/10.1007/b83622>
- [2] H. Samet and M. Tamminen, "Bintrees, csg trees, and time," *SIGGRAPH Comput. Graph.*, vol. 19, no. 3, pp. 121–130, Jul. 1985. [Online]. Available: <http://doi.acm.org/10.1145/325165.325211>
- [3] R. Casati and A. C. Varzi, *Parts and places: The structures of spatial representation*. MIT Press, 1999. [Online]. Available: <http://dx.doi.org/10.1215/00318108-110-3-479>
- [4] S. Abiteboul, R. Hull, and V. Vianu, *Foundations of databases*. Addison-Wesley Reading, 1995, vol. 8.
- [5] A. G. Cohn, B. Bennett, J. Gooday, and N. M. Gotts, "Qualitative spatial representation and reasoning with the region connection calculus," *Geoinformatica*, vol. 1, no. 3, pp. 275–316, 1997. [Online]. Available: <http://dx.doi.org/10.1023/A:1009712514511>
- [6] C.-Y. Huang and K.-L. Chung, "Fast operations on binary images using interpolation-based bintrees," *Pattern Recognition*, vol. 28, no. 3, pp. 409–420, 1995. [Online]. Available: [http://dx.doi.org/10.1016/0031-3203\(94\)00102-r](http://dx.doi.org/10.1016/0031-3203(94)00102-r)
- [7] Y. E. Ioannidis and R. Ramakrishnan, "Efficient transitive closure algorithms," in *VLDB*, vol. 88, 1988, pp. 382–394. [Online]. Available: [http://dx.doi.org/10.1016/0020-0190\(94\)90128-7](http://dx.doi.org/10.1016/0020-0190(94)90128-7)
- [8] R. Agrawal, A. Borgida, and H. V. Jagadish, "Efficient management of transitive relationships in large data and knowledge bases," vol. 18, no. 2, 1989. [Online]. Available: <http://dx.doi.org/10.1145/66926.66950>
- [9] J. F. Allen, "Maintaining knowledge about temporal intervals," *Communications of the ACM*, vol. 26, no. 11, pp. 832–843, 1983. [Online]. Available: <http://dx.doi.org/10.1016/b978-1-4832-1447-4.50033-x>
- [10] K. Aizawa and S. Tanaka, "A constant-time algorithm for finding neighbors in quadtrees," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 31, no. 7, pp. 1178–1183, 2009. [Online]. Available: <http://dx.doi.org/10.1109/tpami.2008.145>

On (in)Validity of Aristotle's Syllogisms Relying on Rough Sets

Tamás Kádek

University of Debrecen

Faculty of Informatics

Kassai street 26, 4028 Debrecen, Hungary

Email: kadek.tamas@inf.unideb.hu

Tamás Mihálydeák

University of Debrecen

Faculty of Informatics

Kassai street 26, 4028 Debrecen, Hungary

Email: mihalydeak.tamas@inf.unideb.hu

Abstract—The authors investigate the properties of first-order logic having its semantics based on a generalized (partial) approximation of sets. The goal of the investigation in this article is to compare the classical first-order semantics with a partial and lower approximation-based one. The idea is that lower approximation represents the reliable knowledge, so the reasoning used by the lower approximation may be valid or may be valid with some limitations. First, the authors show an experimental result which confute the previous supposition and the result of an algorithm which generates refutations for some well-known valid arguments: the 12 syllogisms of Aristotle. We think that these syllogisms represent the most common usage of categorical statements. A language with single-level quantification is constructed, as syllogisms can be formalized using this language. Based on the experimental results, the authors suggest some modifications of the semantics if the goal is to approximate the classical case.

I. INTRODUCTION

THE rough set theory gives the ability to construct different first-order logical systems (see [1], [2]). By the generalization¹ of rough set theory, the truth domain of a formula can be approximated using a partial approximation of sets. The authors introduced earlier a tool-based system as the semantical basis of a generalized first-order logic [5]. The introduced language let us use more than one kind of approximation and allowed it in the language level — with approximative sentence functors — to mix the crisp and rough evaluation. This rich language was very different from the classical case; furthermore, many of the classical rules — such as modus-ponens — failed when they were combined with the approximative functors. In this work, we focus only to the lower approximation of sets (later the lower approximation of truth domains) because of the naïve idea that while the upper approximation represents possibility, the lower approximation represents certainty.

We are eager to know whether what we conclude using the approximation is equal to what we conclude using the classical first-order reasoning. Whether it is possible to formulate some conditions which guarantee the validity of the results made by the approximation.

¹Different generalizations of rough set theory (see [3]) and granular computing play a crucial role in computer sciences (see, e.g. in [4]).

The investigation starts with an experiment, testing some well-known valid arguments, the 12 syllogisms of Aristotle. These syllogisms were chosen in order to represent the most common usage of categorical statements. In the experiment, we use a simplified language which gives us the ability to formalize the syllogisms and test their validity using a lower approximation-based semantics. The language is restricted to single-level quantification only, but it is still expressive enough to formalize categorical statements. Initially, this semantics is defined in the same way as in the tool-based first-order case, but restricted only to the lower approximation. Later, we suggest some modifications in the semantical level to ensure the validity of the classical arguments.

II. ARISTOTLE'S SYLLOGISMS

Aristotle's syllogisms are valid reasoning, constructed from three sentences: two premises and one conclusion. The premises are usually categorized into four types: [6]

- *a*-type $\forall x(p_1(x) \supset p_2(x))$
- *i*-type $\exists x(p_1(x) \wedge p_2(x))$
- *e*-type $\neg \exists x(p_1(x) \wedge p_2(x))$
- *o*-type $\exists x(p_1(x) \wedge \neg p_2(x))$

Each statement contains two from three predicates — usually denoted by p, s, m — and each predicate appears in exactly two statements.

1st figure	2nd figure	3rd figure	
$m-p$	$p-m$	$m-p$	premise
$s-m$	$s-m$	$m-s$	premise
$s-p$	$s-p$	$s-p$	conclusion

For example, the syllogism called *Barbara* contains only *a*-type premises and *a*-type conclusion:

$$\forall x(m(x) \supset p(x)), \forall x(s(x) \supset m(x)) \models \forall x(s(x) \supset p(x))$$

During the investigation, we focus on the syllogisms from the first 3 figures, and we do not take care of those which require an existential pre-supposition, only the remaining 12:

- 1st figure: *Barbara, Celarent, Darii, Ferio*
- 2nd figure: *Cesare, Camestres, Festion, Baroco*
- 3rd figure: *Disamnis, Datisi, Bocardo, Ferison*

There were several similarities in the results, that is why this article presents only those which belong to the first figure. All

of them are valid in the classical case, and now the question is: could we create any refutations using lower approximations only?

III. SIMPLIFIED PARTIAL FIRST-ORDER LOGIC BASED ON SET APPROXIMATION

In this section, we would like to introduce a simplified first-order language, expressive enough to formalize the syllogisms.

A. First-Order Language

Let $\langle LC, Var, Con, Tool, Form \rangle$ be a simplified first-order language, where:

- logical constant symbols $LC = \{\neg, \wedge, \supset, \exists, \forall, (,)\}$,
- variables $Var = \{x\}$, note that one variable is enough,
- nonlogical constant symbols $Con = \{p_1, p_2, \dots, p_n\}$, where $n \geq 1$,
- set of tools $Tool = \{t_1, t_2, \dots, t_k\}$, where $k \geq 1$.

The formulas of the language are given by the following definition:

- 1) Let QF be the set of quantification-free expressions, so that
 - a) $p(x) \in QF$ — and it is atomic — if $p \in Con$,
 - b) $\neg A \in QF$ if $A \in QF$,
 - c) $(A \wedge B) \in QF$ and $(A \supset B) \in QF$ if both $A \in QF$ and $B \in QF$
- 2) The set $Form$ is given by the following inductive definition:
 - a) $\forall x A \in Form$ and $\exists x A \in Form$ if $A \in QF$
 - b) $\neg A \in Form$ if $A \in Form$,
 - c) $(A \wedge B) \in Form$ and $(A \supset B) \in Form$ if both $A \in Form$ and $B \in Form$

Note that the language is restricted to unary predicates (that is, they are monadic) and single-level quantification. So we define a fragment of the first-order logic which is decidable. The disjunction symbol \vee is also missing from the language, it is not necessary to formalize the syllogisms later, but this is not a real restriction.

B. Partial Approximation of Sets

The ordered 5-tuple $\langle U, \mathfrak{B}, \mathfrak{D}_{\mathfrak{B}}, l, u \rangle$ is a general partial approximation space² if

- 1) U is a nonempty set;
- 2) $\mathfrak{B} \subseteq 2^U \setminus \emptyset$, $\mathfrak{B} \neq \emptyset$;
- 3) $\mathfrak{D}_{\mathfrak{B}}$ is an extension of \mathfrak{B} , i.e., $\mathfrak{B} \subseteq \mathfrak{D}_{\mathfrak{B}}$, such that $\emptyset \in \mathfrak{D}_{\mathfrak{B}}$; and $\bigcup B \in \mathfrak{D}_{\mathfrak{B}}$ for all $B \subseteq \mathfrak{B}$
- 4) the functions l and u form a Pawlakian approximation pair $\langle l, u \rangle$, i.e.,
 - a) the lower approximation of an $S \in 2^U$ set is

$$l(S) \stackrel{def}{=} \bigcup \{B : B \in \mathfrak{B} \text{ and } B \subseteq S\};$$

- b) the upper approximation of an $S \in 2^U$ set is

$$u(S) \stackrel{def}{=} \bigcup \{B : B \in \mathfrak{B} \text{ and } B \cap S \neq \emptyset\}.$$

²One of the most general notion of weak and strong approximation pairs can be found in Dütsch and Gediga [7].

The Pawlakian approximation pair was chosen because it is very well-known and most widely used. For other solutions see [8].

C. Interpretation

The $\langle U, \varrho \rangle$ pair is an interpretation of the language $\langle LC, Var, Con, Tool, Form \rangle$ if

- U is a nonempty set of objects, and
- $\varrho : Con \cup Tool \rightarrow 2^U$ is a mapping, and
- $\varrho(t_i) \neq \emptyset$ for all $i \in \{1, \dots, k\}$.

The ϱ mapping assigns a truth domain to each nonlogical constant symbol and a nonempty truth domain to the tools.

D. Semantic Rules

Let $\langle U, \mathfrak{B}, \mathfrak{D}_{\mathfrak{B}}, l, u \rangle$ be a general partial approximation space generated by the $\langle U, \varrho \rangle$ interpretation of a given $\langle LC, Var, Con, Tool, Form \rangle$ simplified first-order language. The ϱ mapping and the $Tool$ set generate the approximation space:

$$\mathfrak{B} = \{\varrho(t) : t \in Tool\}$$

No sentence functors appears in the language level. But the semantic value of a formula $\llbracket F \rrbracket^{(U, \varrho)}$ and the semantic value of the quantification-free expressions $\llbracket Q \rrbracket_{x \mapsto u}^{(U, \varrho)}$ are defined based on the lower approximation.

Let $\overset{w}{\neg}, \overset{w}{\supset}, \overset{w}{\wedge}$ be weak Kleene connectives [9], such that

$\overset{w}{\neg}$		$\overset{w}{\supset}$	0	1	2	$\overset{w}{\wedge}$	0	1	2
0	1	0	1	1	2	0	0	0	2
1	0	1	0	1	2	1	0	1	2
2	2	2	2	2	2	2	2	2	2

Our selection fell on Kleene's weak connectives because of the idea to keep the truth value gap. It was the basis of the semantics defined later for quantifiers too.

- 1) The semantic value of an atomic expression $p(x) \in QF$ using a given interpretation $\langle U, \varrho \rangle$ and a variable assignment $x \mapsto u$ where $u \in U$:

$$\llbracket p(x) \rrbracket_{x \mapsto u}^{(U, \varrho)} \stackrel{def}{=} \begin{cases} 1 & \text{if } u \in l(\varrho(p)) \\ 0 & \text{if } u \in l(U \setminus u(\varrho(p))) \\ 2 & \text{otherwise} \end{cases} \quad (1)$$

where the $\langle l, u \rangle$ approximation pair belongs to the approximation space generated by the $Tool$ and $\langle U, \varrho \rangle$.

- 2) The semantic value of quantification-free expression is defined recursively

$$\begin{aligned} \llbracket \neg A \rrbracket_{x \mapsto u}^{(U, \varrho)} &\stackrel{def}{=} \overset{w}{\neg} \llbracket A \rrbracket_{x \mapsto u}^{(U, \varrho)} \\ \llbracket (A \supset B) \rrbracket_{x \mapsto u}^{(U, \varrho)} &\stackrel{def}{=} \llbracket A \rrbracket_{x \mapsto u}^{(U, \varrho)} \overset{w}{\supset} \llbracket B \rrbracket_{x \mapsto u}^{(U, \varrho)} \\ \llbracket (A \wedge B) \rrbracket_{x \mapsto u}^{(U, \varrho)} &\stackrel{def}{=} \llbracket A \rrbracket_{x \mapsto u}^{(U, \varrho)} \overset{w}{\wedge} \llbracket B \rrbracket_{x \mapsto u}^{(U, \varrho)} \end{aligned}$$

- 3) The semantic value of a formula from $Form$

TABLE I
ARISTOTLE'S SYLLOGISMS — THE FIRST FIGURE

Syllogism	first premise	second premise	conclusion
<i>Barbara</i>	<i>a</i> -type m, p $\forall x(m(x) \supset p(x))$	<i>a</i> -type s, m $\forall x(s(x) \supset m(x))$	<i>a</i> -type s, p $\forall x(s(x) \supset p(x))$
<i>Celarent</i>	<i>e</i> -type m, p $\neg \exists x(m(x) \wedge p(x))$	<i>a</i> -type s, m $\forall x(s(x) \supset m(x))$	<i>e</i> -type s, p $\neg \exists x(s(x) \wedge p(x))$
<i>Darii</i>	<i>a</i> -type m, p $\forall x(m(x) \supset p(x))$	<i>i</i> -type s, m $\exists x(s(x) \wedge m(x))$	<i>i</i> -type s, p $\exists x(s(x) \wedge p(x))$
<i>Ferio</i>	<i>e</i> -type m, p $\neg \exists x(m(x) \wedge p(x))$	<i>i</i> -type s, m $\exists x(s(x) \wedge m(x))$	<i>o</i> -type s, p $\exists x(s(x) \wedge \neg p(x))$

$$\llbracket \forall x A \rrbracket^{(U, \varrho)} \stackrel{\text{def}}{=} \begin{cases} 2 & \text{if } \llbracket A \rrbracket_{x \mapsto u}^{(U, \varrho)} = 2 \text{ for all } u \in U, \\ 0 & \text{if there is an } u \in U, \\ & \text{where } \llbracket A \rrbracket_{x \mapsto u}^{(U, \varrho)} = 0, \\ 1 & \text{otherwise.} \end{cases}$$

$$\llbracket \exists x A \rrbracket^{(U, \varrho)} \stackrel{\text{def}}{=} \begin{cases} 2 & \text{if } \llbracket A \rrbracket_{x \mapsto u}^{(U, \varrho)} = 2 \text{ for all } u \in U, \\ 1 & \text{if there is an } u \in U, \\ & \text{where } \llbracket A \rrbracket_{x \mapsto u}^{(U, \varrho)} = 1, \\ 0 & \text{otherwise.} \end{cases}$$

where $A \in \text{Form}$.

$$\begin{aligned} \llbracket \neg A \rrbracket^{(U, \varrho)} &\stackrel{\text{def}}{=} \neg \llbracket A \rrbracket^{(U, \varrho)} \\ \llbracket (A \supset B) \rrbracket^{(U, \varrho)} &\stackrel{\text{def}}{=} \llbracket A \rrbracket^{(U, \varrho)} \supset \llbracket B \rrbracket^{(U, \varrho)} \\ \llbracket (A \wedge B) \rrbracket^{(U, \varrho)} &\stackrel{\text{def}}{=} \llbracket A \rrbracket^{(U, \varrho)} \wedge \llbracket B \rrbracket^{(U, \varrho)} \end{aligned}$$

where $A, B \in \text{Form}$.

IV. EXPERIMENTAL RESULTS

If we have $|U| = 4$, then the number of different interpretations is:

$$(2^{|U|})^3 \cdot (2^{(2^{|U|}-1)} - 1) = 4096 \cdot 32767 = 134213632,$$

where the number of different approximation spaces is 32767 (where the members of the *Tool* set has different $\varrho(t)$ nonempty truth domain), and there exists 4096 different interpretation for $\text{Con} = \{p, s, m\}$. With such a small U , there is an efficient way to implement the formula evaluation. The idea is that if there is a given $\langle U, \varrho \rangle$ interpretation, we can generate the truth domain and falsity domain for each predicate before the evaluation.

Let us define the truth and falsity domain of an atomic formula in the classical case

$$[p]^+ = \varrho(p) \text{ and } [p]^- = U \setminus \varrho(p) \text{ if } p \in \text{Con} \cup \text{Tool}$$

and in case of the introduced semantics (with the lower approximation based on (1))

$$\begin{aligned} [\downarrow p]^+ &\stackrel{\text{def}}{=} \text{l}(\varrho(p)) \\ &= \left\{ u \in U : \llbracket p(x) \rrbracket_{x \mapsto u}^{(U, \varrho)} = 1 \right\} \end{aligned}$$

$$\begin{aligned} [\downarrow p]^- &\stackrel{\text{def}}{=} \text{l}(U \setminus \text{u}(\varrho(p))) \\ &= \left\{ u \in U : \llbracket p(x) \rrbracket_{x \mapsto u}^{(U, \varrho)} = 0 \right\} \end{aligned}$$

While the set $[p]^+$ denotes the truth domain of p in case of classical semantics, the $[\downarrow p]^+$ represents the lower approximation of this truth domain. Note that $[\downarrow p]^+ \subseteq [p]^+$ and $[\downarrow p]^- \subseteq [p]^-$. Earlier — in [5] — we defined not only a lower approximation-based semantics but also a first-order language with lower and upper approximative sentence functors denoted by \downarrow and \uparrow . Now we focus only on the lower approximation, supposing that is represents certainty.

The Java code sample shows the implementation of the semantics in the case of the \forall quantifier and in the case of the \wedge connective. The syllogisms are transformed into a postfix form. For example, in case of *Barbara*:

$$\forall x(m(x) \supset p(x)), \forall x(s(x) \supset m(x)) \models \forall x(s(x) \supset p(x))$$

is valid, and so

$$\forall x(m(x) \supset p(x)) \wedge \forall x(s(x) \supset m(x)) \wedge \neg \forall x(s(x) \supset p(x))$$

is unsatisfiable. The last formula is converted to a string representation as "mp>A&m>A&sp>A-&", where \supset , \wedge , \neg , and \forall is replaced with $>$, $\&$, $-$, and A , respectively.

The algorithm uses a pair of stacks to evaluate the postfix expression, one (*sfd* integer array) for falsity domain, and another (*std* integer array) for truth domain. *sp* refers to the top of the stack. The $[\downarrow p]^+$ and $[\downarrow p]^-$ sets are represented by the *ptd*['p'] and *pfid*['p'] integers. Each set data structure is represented as a bit array stored in an integer.

A. Refutations for the First Figure

The table (cf. table II) summarizes the results showing the number of refutations for the syllogisms. It was not necessary

TABLE II
NUMBER OF REFUTATIONS FOR THE FIRST FIGURE

	Number of interpretations	Number of approximation spaces	Barbara	Celarent, Darii, Ferio
Total	134 213 632	32 767	121 536	227 232
$ Tool \leq 3$	2 355 200	575	4 728	9 576
$ Tool \leq 2$	491 520	120	912	912
covering	132 288 512	32 297	117 696	219 936
not covering	1 925 120	470	3 840	7 296
disjoint tools	208 896	51	1 104	1 104
$[p]^+ \neq \emptyset \quad \forall p \in Con$	64 139 967	32 767	110 448	193 776
$[p]^- \neq \emptyset \quad \forall p \in Con$	689 831	32 767	12 168	25 716
$[p]^+ \neq \emptyset \wedge [p]^- \neq \emptyset \quad \forall p \in Con$	229 940	2 680	1368	10 608
$\varrho(p) \neq \varrho(t) \quad \forall t \in Tool, \forall p \in Con$	5 18 720	30 580	540	4 752

Algorithm 1 Calculating the truth value of a formula in Java

```

switch (next) { // next char of postfix formula
case 'A': // forall
    sp--; // pop an argument
    afd = sfd[sp];
    atd = std[sp];
    // push the result
    sfd[sp] = (afd & mask) != 0 ? mask : 0;
    std[sp] = (atd & mask) != 0
        && (afd & mask) == 0 ? mask : 0;
    sp++;
    break;

case '&': // conjunction
    sp--; // pop an argument
    int rfd = sfd[sp];
    int rtd = std[sp];
    sp--; // pop an argument
    int lfd = sfd[sp];
    int ltd = std[sp];
    // push the result
    sfd[sp] = (lfd & (rfd | rtd))
        | (rfd & (lfd | ltd));
    std[sp] = ltd & rtd;
    sp++;
    break;

// ... other connectives ...

default: // an atom
    sfd[sp] = pfd[next];
    std[sp] = ptd[next];
    sp++;
    break;
}

```

to show all the 12 (only those which belong to the first figure), because the same number of refutations appears in the cases of *Barbara*, *Baroco*, and *Bocardo*, as well as in all of the other cases (*Celarent*, *Darii*, *Ferio*, *Cesare*, *Camestres*, *Festio*, *Disamnis*, *Datisi*, and *Ferison*). The upper half of the table summarizes the tested conditions defined on the approximation space. The restrictions on the *Tool* set take their effect on the generated approximation space:

- by restricting the number of different tools to at most 3,
- or at most 2,
- using a covering approximation space $U = \bigcup_{t \in Tool} t$,
- or even a noncovering one,
- or by using tools with disjoint truth domain only.

The outcome achieved is not exactly we had hoped for. The number of approximation spaces decreases, but there still exist some interpretations where the syllogisms do not hold.

In the lower half of the table, there are some restrictions on the interpretation of the predicates (the members of the *Con* set). Here, the number of approximation spaces can be lower than 32 767 only if there are some approximation spaces where none of the interpretations satisfy the condition:

- the truth domain of the lower approximated predicates must not be empty,
- the falsity domain of the lower approximated predicates must not be empty,
- neither the truth domain nor the falsity domain of the lower approximation of the predicates are empty,
- there is no predicate which is also a tool.

	p	s	m	t_1	t_2	t_3	$\downarrow p$	$\downarrow s$	$\downarrow m$
u_1	0	0	0	1	0	0	0	0	0
u_2	1	0	0	0	1	0	1	0	0
u_3	0	1	1	0	0	1	2	1	1
u_4	1	1	1	0	0	1	2	1	1

The above example shows an interpretation which is a refutation for all of the followings: *Darii*, *Ferio*, *Festio*, *Datisi*, and *Ferison*. Note that the approximation space is covering, each approximated predicate has nonempty truth domain and nonempty falsity domain. The column title p is an abbreviation for $|p(x)|_{x \mapsto u}^{(U, \varrho)}$, and $\downarrow p$ is for $\llbracket p \rrbracket_{x \mapsto u}^{(U, \varrho)}$.

B. How to Create Refutations?

The presented experimental results show that the lower approximation and the three-valued — two-valued but with truth value gap — semantics could not represent irrefutable

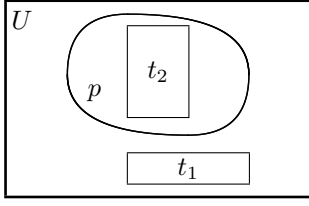


Fig. 1. Illustration of an approximation space

knowledge. In this section, we show a simple way to construct a formula which is easy to refute.

Now our goal is to construct a formula A such that $\llbracket A \rrbracket = 1$ but $|A| = 0$. It is enough to create a sentence which is about objects that exist but are outside of the lower approximation.

Let $P = \{p_1, \dots, p_n\}$ and $N = \{p_{n+1}, \dots, p_{n+m}\}$ be disjoint sets of predicates, such that $n + m \geq 1$. Let $S \subseteq U$ be a set, where

- 1) $S = \left(\bigcap_{i=1}^n [p_i]^+ \right) \cap \left(\bigcap_{i=n+1}^m [p_i]^- \right) \neq \emptyset$
- 2) $S \cap \left(\bigcap_{i=1}^n [\downarrow p_i]^+ \right) \cap \left(\bigcap_{i=n+1}^m [\downarrow p_i]^- \right) = \emptyset$
- 3) $\bigcap_{i=1}^{n+m} ([\downarrow p_i]^+ \cup [\downarrow p_i]^-) \neq \emptyset$

Since the set S is not empty,

$$|\neg \exists x (p_1(x) \wedge \dots \wedge p_n(x) \wedge \neg p_{n+1}(x) \wedge \dots \wedge \neg p_{n+m}(x))| = 0,$$

but — because of the second criterion — the set is hidden for the lower approximation. It causes

$$\llbracket \neg \exists x (p_1(x) \wedge \dots \wedge p_n(x) \wedge \neg p_{n+1}(x) \wedge \dots \wedge \neg p_{n+m}(x)) \rrbracket \neq 0.$$

The third criterion ensures computability. In other words, it avoids the truth value gap, so the existentially quantified formula must have a truth value other than 2. As a result,

$$\llbracket \neg \exists x (p_1(x) \wedge \dots \wedge p_n(x) \wedge \neg p_{n+1}(x) \wedge \dots \wedge \neg p_{n+m}(x)) \rrbracket = 1.$$

C. Example

Let $\langle U, \varrho \rangle$ be an interpretation — illustrated by Fig. 1 — for a language with $Con = \{p, t_1, t_2\}$ and $Tool = \{t_1, t_2\}$.

Let S be a set in connection with the approximation space such that $P = \{p\}$ and $N = \{t_2\}$.

- 1) $S = [p]^+ \cap [t_2]^-$ and $S \neq \emptyset$
- 2) $S \cap [\downarrow p]^+ \cap [\downarrow t_2]^- = S \cap [t_2]^+ \cap [t_1]^+ = \emptyset$
- 3) $([\downarrow p]^+ \cup [\downarrow p]^-) \cap ([\downarrow t_2]^+ \cup [\downarrow t_2]^-) = ([t_2]^+ \cup [t_1]^+) \cap ([t_2]^+ \cup [t_1]^+) \neq \emptyset.$

The formula created from the sets P and N is

$$\neg \exists x (p(x) \wedge \neg t_2(x)),$$

which is false in the classical case. There are some objects in $S = \varrho(p) \cap (U \setminus \varrho(t_2)) \neq \emptyset$, but the lower approximation hides them. Even in the very simple case, which is illustrated by 1, it was easy to show that $|A| = 1$ is not the logical consequence of $\llbracket A \rrbracket = 1$.

A sample $\langle U, \varrho \rangle$ interpretation in a connection with Fig. 1. can be

$$\varrho(p) = \{u_2, u_3\}, \quad \varrho(t_1) = \{u_4\}, \quad \varrho(t_2) = \{u_3\}$$

	p	t_1	t_2	$\downarrow p$	$\downarrow t_1$	$\downarrow t_2$
u_1	0	0	0	2	2	2
u_2	1	0	0	2	2	2
u_3	1	0	1	1	0	1
u_4	0	1	0	0	1	0

in case of $U = \{u_1, u_2, u_3, u_4\}$.

V. RELEVANCE OF THE LOWER APPROXIMATION

In this section, we summarize the conditions of stating that $|A| = 1$ is a logical consequence of $\llbracket A \rrbracket = 1$. The conditions are formalized in a form of fractions, like,

$$\frac{\llbracket A \rrbracket = 1 - \tau \Rightarrow |A| = 1 - \tau}{\llbracket \neg A \rrbracket = \tau \Rightarrow |\neg A| = \tau}$$

where $\tau \in \{0, 1\}$. Note that τ refers to the classical semantical value of a formula A , denoted by $|A|^{(U, \varrho)}$ or for the sake of simplicity: $|A|$. ($|A|^{(U, \varrho)} \in \{0, 1\}$.) To satisfy the condition below the line, it is enough to satisfy the condition above the line. As an example, $\llbracket \neg p(x) \rrbracket = 1 \Rightarrow |\neg p(x)| = 1$ holds if $\llbracket p(x) \rrbracket = 0 \Rightarrow |p(x)| = 0$ holds as well. The \emptyset represents no conditions.

$$\frac{\emptyset}{\llbracket p(x) \rrbracket = \tau \Rightarrow |p(x)| = \tau}$$

$$\frac{\llbracket A \rrbracket = \tau \Rightarrow |A| = \tau; \quad \llbracket B \rrbracket = \tau \Rightarrow |B| = \tau}{\llbracket (A \wedge B) \rrbracket = \tau \Rightarrow |(A \wedge B)| = \tau}$$

$$\frac{\llbracket A \rrbracket = 1 - \tau \Rightarrow |A| = 1 - \tau; \quad \llbracket B \rrbracket = \tau \Rightarrow |B| = \tau}{\llbracket (A \supset B) \rrbracket = \tau \Rightarrow |(A \supset B)| = \tau}$$

The rules above are promising in the case of the zero-order logical connectives but not in the case of the quantifiers.

$$\frac{\llbracket A \rrbracket = 0 \Rightarrow |A| = 0}{\llbracket \forall x A \rrbracket = 0 \Rightarrow |\forall x A| = 0}$$

$$\frac{\llbracket A \rrbracket = 1 \Rightarrow |A| = 1}{\llbracket \exists x A \rrbracket = 1 \Rightarrow |\exists x A| = 1}$$

Using the semantics which gives us the ability to assign 0 or 1 truth value to quantified formulas if we have at least partial information causes that we lost the guarantees to have the classical results.

Now we suggest a modification on the first-order semantics, in a case where only the lower approximation is used. The

*pessimistic*³ semantics of quantifiers is based on the idea that the missing knowledge (represented by the value 2) could be relevant.

The *pessimistic* semantic value of a formula $\llbracket F \rrbracket^{\langle U, \varrho \rangle}$ and the *pessimistic* semantic value of the quantification-free expressions $\llbracket Q \rrbracket^{\langle U, \varrho \rangle}_{x \mapsto u}$ are defined recursively:

- 1) The semantic value of an atomic expression $p(x) \in QF$ using a given interpretation $\langle U, \varrho \rangle$ and a variable substitution $x \mapsto u$ where $u \in U$:

$$\llbracket p(x) \rrbracket^{\langle U, \varrho \rangle}_{x \mapsto u} = \llbracket p(x) \rrbracket^{\langle U, \varrho \rangle}_{x \mapsto u}$$

where the $\langle \cdot, \cdot \rangle$ approximation pairs belongs to the approximation space generated by the *Tool* and $\langle U, \varrho \rangle$.

- 2) The semantic value of a quantified expression is defined as

$$\llbracket \forall x A \rrbracket^{\langle U, \varrho \rangle} \stackrel{def}{=} \begin{cases} 0 & \text{if there is an } u \in U, \\ & \text{where } \llbracket A \rrbracket^{\langle U, \varrho \rangle}_{x \mapsto u} = 0, \\ 1 & \text{if } \llbracket A \rrbracket^{\langle U, \varrho \rangle}_{x \mapsto u} = 1 \text{ for all } u \in U, \\ 2 & \text{otherwise.} \end{cases}$$

$$\llbracket \exists x A \rrbracket^{\langle U, \varrho \rangle} \stackrel{def}{=} \begin{cases} 0 & \text{if } \llbracket A \rrbracket^{\langle U, \varrho \rangle}_{x \mapsto u} = 0 \text{ for all } u \in U, \\ 1 & \text{if there is an } u \in U, \\ & \text{where } \llbracket A \rrbracket^{\langle U, \varrho \rangle}_{x \mapsto u} = 1, \\ 2 & \text{otherwise.} \end{cases}$$

- 3) The semantic value of a quantification-free expression or a formula based on zero-order connectives

Let $\overset{s}{\neg}, \overset{s}{\supset}, \overset{s}{\wedge}$ be strong Kleene connectives, such that

$\overset{s}{\neg}$		$\overset{s}{\supset}$	0	1	2	$\overset{s}{\wedge}$	0	1	2
0	1	0	1	1	1	0	0	0	0
1	0	1	0	1	2	1	0	1	2
2	2	2	2	1	2	2	0	2	2

$$\begin{aligned} \llbracket \neg A \rrbracket^{\langle U, \varrho \rangle}_{x \mapsto u} &\stackrel{def}{=} \overset{s}{\neg} \llbracket A \rrbracket^{\langle U, \varrho \rangle}_{x \mapsto u} \\ \llbracket (A \supset B) \rrbracket^{\langle U, \varrho \rangle}_{x \mapsto u} &\stackrel{def}{=} \llbracket A \rrbracket^{\langle U, \varrho \rangle}_{x \mapsto u} \overset{s}{\supset} \llbracket B \rrbracket^{\langle U, \varrho \rangle}_{x \mapsto u} \\ \llbracket (A \wedge B) \rrbracket^{\langle U, \varrho \rangle}_{x \mapsto u} &\stackrel{def}{=} \llbracket A \rrbracket^{\langle U, \varrho \rangle}_{x \mapsto u} \overset{s}{\wedge} \llbracket B \rrbracket^{\langle U, \varrho \rangle}_{x \mapsto u} \end{aligned}$$

where $A, B \in QF$, and

$$\begin{aligned} \llbracket \neg A \rrbracket^{\langle U, \varrho \rangle} &\stackrel{def}{=} \overset{s}{\neg} \llbracket A \rrbracket^{\langle U, \varrho \rangle} \\ \llbracket (A \supset B) \rrbracket^{\langle U, \varrho \rangle} &\stackrel{def}{=} \llbracket A \rrbracket^{\langle U, \varrho \rangle} \overset{s}{\supset} \llbracket B \rrbracket^{\langle U, \varrho \rangle} \\ \llbracket (A \wedge B) \rrbracket^{\langle U, \varrho \rangle} &\stackrel{def}{=} \llbracket A \rrbracket^{\langle U, \varrho \rangle} \overset{s}{\wedge} \llbracket B \rrbracket^{\langle U, \varrho \rangle} \end{aligned}$$

where $A, B \in Form$.

The goal of using the *pessimistic* semantic in case of the first-order connectives (quantifiers) are the validity of the rules

$$\frac{\llbracket A \rrbracket = 1 \Rightarrow \llbracket A \rrbracket = 1}{\llbracket \forall x A \rrbracket = 1 \Rightarrow \llbracket \forall x A \rrbracket = 1}$$

³Our approach differs from the idea presented in [10] which talks about pessimistic, optimistic, and average membership functions.

$$\frac{\llbracket A \rrbracket = 0 \Rightarrow \llbracket A \rrbracket = 0}{\llbracket \exists x A \rrbracket = 0 \Rightarrow \llbracket \exists x A \rrbracket = 0}$$

Unfortunately, as another effect, it increases the number of interpretations, where a quantified formula has truth value gap. Kleene's strong connectives have opposite effect. It is not necessary to change the weak connectives to strong, the goal is reached anyway:

$$\llbracket A \rrbracket = \tau \Rightarrow \llbracket A \rrbracket = \tau.$$

VI. CONCLUSION

As a general observation, we can conclude that if we change the semantics to lower approximation-based, then the syllogisms of Aristotle are not valid. There are several ways for further investigation by creating restrictions on the approximation space or by changing the semantical meaning of the logical connectives or the first-order quantifiers, as it was demonstrated in the second half of the article.

ACKNOWLEDGEMENTS

Many thanks to the anonymous referees for their valuable suggestions.

REFERENCES

- [1] Chakraborty, M.K., Banerjee, M., "Rough logic with rough quantifiers," Bull. Polish Acad. Sc. (Math.), **41**, 4, pp. 305–315, 1993
- [2] Banerjee, M., Chakraborty, M.K.: "Rough logics: a survey with further directions," in: *Incomplete Information: Rough Set Analysis*, Orłowska, E. (ed.), Physica, Springer-Verlag, Heidelberg, pp. 579–600, 1998
- [3] Yao, Y.Y.: "On generalizing rough set theory," *Proceedings of the 9th International Conference Rough Sets, Fuzzy Sets, Data Mining, and Granular Computing (RSFDGrC 2003)*. LNAI 2639, Springer-Verlag, pp. 44–51, 2003
- [4] Yao, Y.Y.: "Information granulation and rough set approximation," *International Journal of Intelligent Systems* **16** (1), pp. 87–104, 2001
- [5] Kádek, T., Mihálydeák, T.: "Some Fundamental Laws of Partial First-Order Logic Based on Set Approximations," in *Rough Sets and Current Trends in Computing*, Lecture Notes in Computer Science Volume 8536, Cornelis, C., Kryszkiewicz, M., Slezak, D., Menasalvas Ruiz, E., Bello, R., Shang, L. (eds.) 9th International Conference, RSCTC 2014, Granada and Madrid, Spain, pp. 47–58, 2014
- [6] van Rooij, R.: "The propositional and relational syllogistic," *Logique et Analyse*, **55**, pp. 85–108, 2012
- [7] Düntsch, I., Gediga, G.: "Approximation operators in qualitative data analysis," in *Theory and Applications of Relational Structures as Knowledge Instruments*. Volume 2929 of Lecture Notes in Computer Science, de Swart, H.C.M., Orłowska, E., Schmidt, G., Roubens, M., (eds.) Springer, pp. 214–230, 2003
- [8] Csajbók, Z., Mihálydeák, T.: "Fuzziness in Partial Approximation Framework," in *Annals of Computer Science and Information Systems*, Maria Ganzha, Leszek Maciaszek, Marcin Paprzycki (eds.) Volume 1: Proceedings of the 2013 Federated Conference on Computer Science and Information Systems (FedCSIS). pp. 35–41, 2013
- [9] Maierborn C., Heusinger K., Portner P. (eds.): *Semantics*, **3**, 2012, Walter de Gruyter GmbH, Berlin/Boston, ISBN 978-3-11-025337-5
- [10] Mihálydeák, T.: "Partial first-order logic relying on optimistic, pessimistic and average partial membership functions," in *Proceedings of the 8th conference of the European Society for Fuzzy Logic and Technology Pasi, G., Montero, J., Ciucci, D. (eds.) (EUSFLAT 2013)*, 2013 ISBN (on-line): 978-90786-77-78-9 doi:10.2991/eusflat.2013.53

Parthood and Convexity as the Basic Notions of a Theory of Space

Klaus Robering

Institute of Design and Communication

University of Southern Denmark

Kolding, Denmark

Email: robering@sdu.dk

Abstract—A deductive system of geometry is presented which is based on atomistic mereology (“mereology with points”) and the notion of convexity. The system is formulated in a liberal many-sorted logic which makes use of class-theoretic notions without however adopting any comprehension axioms. The geometry developed within this framework roughly corresponds to the “line spaces” known from the literature; cf. [1, p. 155]. The basic ideas of the system are presented in the article’s INTRODUCTION within a historical context. After a brief presentation of the logical and mereological framework adopted, a “pregeometry” is described in which only the notion of convexity but no further axiom is added to that background framework. Pregeometry is extended to the full system in three steps. First the notion of a line segment is explained as the convex hull of the mereological sum of two points. In a second step two axioms are added which describe what it means for a thus determined line segment to be “straight”. In the final step we deal with the order of points on a line segment and define the notion of a line. The presentation of the geometric system is concluded with a brief consideration of the geometrical principles known by the names of Peano and Pasch. Two additional topics are treated in short sections at the end of the article: (1) the introduction of coordinates and (2) the idea of a “geometrical algebra”.

I. INTRODUCTION

Geometry is a very old science and from its very beginnings it was a classical place for discussing the relationship between qualitative and quantitative reasoning. Synthetic geometry as developed in the first four books of Euclid’s *Elements* [2] is a paradigm instance of qualitative reasoning. The concept of number is only introduced in Book VII of the work; and the treatment of numbers and the investigation of their properties make use of geometrical representations. As is testified by our parlance about square and cubic numbers, remnants of this procedure are still present in contemporary mathematical terminology. Zeuthen, in his history of ancient mathematics, refers to this procedure as “geometric arithmetic” and “geometric algebra”; cf. [3, pp. 40–53].

The relationship between geometry and algebra was turned around when, in the 17th century, Fermat and Descartes, translating geometric construction tasks into problems concerning the solution of equations, laid the foundations of analytic geometry and thus paved the way for the use of algebra and, later, calculus for the solution of geometric problems. This is rightly considered a major breakthrough in geometric research. However, not so long after Fermat’s and Descartes’ innovation, already Leibniz argued that the

use of (numerical) analysis for achieving geometric results is a detour since analysis is concerned with magnitude and thus only indirectly (“per circuitum”) faces such geometric notions as shape (“forma”) and similarity (“similitudo”); cf. [4]. According to him, numerical analysis is therefore to be supplemented by a geometric analysis—an *analysis situs*—which deals with such important geometric properties in a direct rather than roundabout way. This geometric analysis is based on a calculus of geometric concepts which makes use of a symbolic language (“characteristica geometrica”) resembling that of algebra. Though “analysis situs” became the original, now obsolete name for what is called (general) topology today, there is scarcely a connection between this modern discipline and Leibniz’ original ideas.¹ However, Hermann Grassmann, in a treatise submitted as an answer to a prize question asked by a scientific society of Leibniz’s hometown Leipzig, re-interpreted Leibniz’s ideas in the framework of his “lineale Ausdehnungslehre”, which we today consider as a rather abstract and general formulation of vector algebra.²

What Leibniz had in mind when he proposed his *analysis situs*, was not a simple return to Euclid’s synthetic method and to his deductive procedure but rather an algebraic formulation of geometry in which one could confirm geometric proofs by calculations which directly deal with such geometric entities such as angles, triangles, squares, and circles without first encoding them into numbers, thus translating a geometric problem into one of numerical algebra or analysis. By this he hoped to replace long and intricate arguments to be found in Euclid’s *Elements* by simple calculations; cf. the examples given by him at the end of this brief note; [4, pp. 181–183]. Today we are tempted to say that he tried to reduce the

¹The reference to “position” (Latin *situs*) is not uncommon in geometric research of the 19th century. Thus, for instance, in 1803 the French mathematician L. N. M. Carnot published a book with the title *Géométrie de position*, in which he tried to combine intuitive synthetic geometry with algebraic analytic geometry. Another example is Ch. von Staudt, who in 1847 presented his formulation of projective geometry in a book *Geometrie der Lage*, which is an exact German translation of the title of Carnot’s book. Leibniz, when developing his idea of a “characteristica geometrica”, might have been acquainted with the geometric ideas of Desargues which led up in the 19th century to the development of projective geometry; cf. [5]. — The 19th century is generally considered “a golden age of geometry”; cf. [6, ch.].

²Both works of Grassmann, his *Lineale Ausdehnungslehre* and his *Geometrische Analyse geknüpft an die von Leibniz erfundene geometrische Charakteristik*, have been reprinted in [7].

computational complexity of spatial reasoning. Grassmann, in the treatise mentioned in the previous paragraph, delivers an insightful analysis of Leibniz' first attempts in that direction, cf. [7, pp. 328–334] and puts forwards, in the framework of his own “Ausdehnungslehre”, some suggestions for improving Leibniz' work. In order to be able to apply his conceptual framework to geometry, however, he analyses “geometric magnitudes” such as points and line segments as pairs consisting of a geometric entity (a position in the case of a “point magnitude” and a direction in the case of a “line magnitude”) and a “metrical value” (“Masswerth”). Hence in his algebraic analysis of Euclid's basic operation of connecting two points by a straight line he re-introduces numbers which Leibniz wished to eliminate; cf. [7, p. 355f].

An analysis of the Euclidean operation of joining two points which is more consonant with Leibniz' original ideas has been given only much later by Walter Prenowitz; cf. [8], [9]. Given (not necessarily different) points p_1 , p_2 , their join p_1p_2 is the linear segment between them; cf. [9, p. 3].³ For the join operation, then, algebraic laws such as that of commutativity $p_1p_2 = p_2p_1$, associativity $(p_1(p_2p_3)) = (p_1p_2)p_3$, and idempotency ($pp = p$) are postulated. This looks as if the set of points and the join operation make up an idempotent, Abelian groupoid. However, a closer look upon the first law reveals that this algebraization of a geometric topic rests upon a notational convention. The result p_2p_3 of joining the points p_2 and p_3 is a line segment; but what then is the result of joining this line segment to a point? Prenowitz conceives of line segments as point sets. The range of the binary join operation thus is a set of point sets; furthermore, he declares $(p_1(p_2p_3))$ to be the set of points lying on segments which join the point p_1 with some point of the set p_2p_3 . Hence $p_1(p_2p_3)$ is the set of points within the triangle $\triangle p_1p_2p_3$. If we want to conform to absolute exactness, we should either conceive of the join operation as an operation on point sets —and thus formulate associativity by something like $\{p_1\} \cdot p_2p_3 = p_1p_2 \cdot \{p_3\}$ —or we should use a background set theory which identifies urelements with their singletons.⁴ Ignoring the distinction between an individual and its singleton set, forces one also to blur the distinction between the relations of membership and inclusion: as an individual the point p_1 is a member of the set p_1p_2 , but as its own singleton it is also a subset of that set.

Prenowitz, of course, is completely aware of this; cf. his footnote 5 in [8, p. 3]. Instead of relying on the good instinct of the reader of his writings who restores the set-theoretic distinctions whenever necessary, there would have been an alternative for him, namely to use mereology instead of set theory as a background theory. Given a mereological background, single points of a line segments bear the same relationship to that segment as complete subsegments do: both its points and its

subsegments are just parts of the entire segment. The present article follows the strategy just suggested by adopting mereology as a framework for geometry. This issue will be taken up in section II-B below. Adopting mereology “homogenizes” points and segments: both are individuals and the arguments of the join operation are thus on an equal footing. Terms like “ $p_1(p_2p_3)$ ” can be interpreted in a straightforward way which does not require special care of the reader. However, mereology does not resolve our problem completely. It is fine to have both points and line segments as first class citizens of the entire universe of discourse, but these two entities are nevertheless of different kinds. There are things we want to say about points which do not make sense for segments. Since the inception of many-sorted logic systems in the 1930s geometry always has been a prime application area for many-sorted logics; and Arnold Schmidt [11, p. 32], in his classical article on this topic, explicitly refers to Hilbert's axiomatisation of Euclidean geometry [12] in order to motivate the introduction of sortal distinctions. Introducing sortal distinctions between points and segments, however, reintroduces our problem with the interpretation of terms like $p_1(p_2p_3)$ —unless, of course, a more liberal sort system is adopted that allows for the crossing of sort boundaries which is strictly forbidden in such rigid systems as that of Schmidt [11]. Such a liberal system, due to Arnold Oberschelp [13, ch. 3], is adopted in section II below.

The main use Prenowitz makes of the join operation is to define the notion of convexity which is central for his approach to geometry; cf. [9, pp. 25–28]. There is plenty of reason to follow Prenowitz in assigning a central role to the notion of convexity. (1) It plays a central role in various other parts of mathematics as documented in the comprehensive handbook [14]. (2) In quite a few important applications of computational geometry it plays a crucial role; cf. the list given in [15, p. 63]. (3) It seems to be of special importance for the human cognitive systems also in areas beyond geometry; cf. [16, pp. 69–74, 157–174].⁵ We shall therefore give convexity a central position in our system of geometry presented below. Its position in that system is even more central than that it occupies in Prenowitz' since we start with the notion of convexity and define that of a linear segment in terms of it whereas definitional dependence in the other way round in Prenowitz' system.⁶ However, first our logical

⁵Given the importance and usefulness of the notion of convexity, it does not come to a big surprise that it already has made its appearance in formal systems for the representation of spatial knowledge; cf., e. g., [17], [18], and [19]. In [17, sec. 4.3] it is assumed that the convexity function $conv$ which assigns to regions their convex hulls “is only well sorted when defined on one piece regions”. No such restriction is assumed here for the hull operator h which will be introduced below in Def 14. The domain of discourse of the interesting theory put forward by [19] is the set of “regular open rational polygons of the real plane” (p. 5). We adopt a much more comprehensive notion of a region (cp. fn. 8) and do not make any decision on the matter of dimensions.

⁶In a strict formal sense, we actually do not define the notion of a segment in terms of convexity. The first notion is present in our system from the start since the many-sorted language used comprises a special sort s of segments. However, the axiom *Mer 3* below specifies a sufficient and necessary condition for being a segment.

³In Prenowitz' 1943 article on this topic, the additive notation “ $p_1 + p_2$ ” is preferred; cf. [8, p. 236]. For Prenowitz, the segment p_1p_2 resulting from joining p_1 and p_2 does not include these two boundary points. Thus, for him, a segment is an “open” set of points. In contrast to this, the join operation which will be defined below in *Def. 14*-(c) includes the boundary points.

⁴As is done, for instance, in Quine's NF; cf. [10].

and mereological background has to be explained in order to prepare the stage for the treatment of these geometric topics.

II. BACKGROUND: CLASS THEORY AND MEREOLOGY

The system of geometry proposed here builds upon two more basic formal theories: (1) a certain system of “class logic” and (2) a system of mereology.⁷ The version of class logic chosen here is the system LC developed by Arnold Oberschelp [13, chap. 3]. Basically, class logic is “set theory without comprehension axioms”. The system LC will be described in more detail in the first part of the present section. Mereology has been used as an ingredient in several axiomatisation of geometric theories; cf., e.g., [23], [24], [25], [26], [27]. The specific system of mereology used here will be introduced in section II-B below.

A. Class Theory

The specific version of LC used here is formulated within a many-sorted language with four sorts denoted by **u**, **c**, **s**, **p**. The universal sort **u** is the sort of all regions. In the present system a region is any mereological sum of points. Thus a region does not need to be connected or three-dimensional, but each region has at least one punctual part; cf. MER 6 below.⁸ The remaining three sorts are, respectively, the sort of convex regions (**c**), of (linear) segments (**s**), and of points (**p**). The universal sort contains all other sorts as its subsorts; segments are special convex regions and points, as we shall see below (cf. The 17), special segments. For each sort there are infinitely many variables. We reserve the letter used as the index of a sort for the variables of that sort; thus, e.g., p, p_1, p_2, \dots are the variables for points. The letters v, w, v_1, v_2, \dots are used as meta-linguistic signs for variables (of any sort); u always refers to a variable of sort **u**. If it is else necessary to indicate a term’s membership in a sort s , this will be done by adding “ s ” as a superscript. Besides the variables there is only one single constant “**P**” for the part-whole-relation; this constant is an example of a class term and is not assigned to any of the sorts. Semantically LC distinguishes between individuals and objects. Individuals are special objects; they are the possible values of the variables hence the “real” objects. Some

objects, however, are not individuals; they are only “virtual”, lie without the domain of quantification, and thus do not belong to any sort. LC abstains from any assumptions about the existence of classes and so class terms may denote merely virtual objects. “**P**” stands for a relation, i.e., a class of pairs of individuals.

There are three groups of logical signs in our version of LC: (1) the connectives $\neg, \wedge, \vee, \rightarrow$, and \leftrightarrow ; (2) the quantifiers \exists and \forall ; (3) the relational signs $=$ (identity) and \in (membership); (4) the elementary term constructor \langle, \rangle (pair formation); and finally (5) the variable binding term constructors \imath (definite description) and $\{ \mid \}$ (class formation). We use the letters “ X ” and “ Y ” as metalinguistic variables for terms, and “ φ ” and “ ψ ” for formulas. These two classes of expressions are defined by a simultaneous recursion. (a) Each variable is a term and so is the constant “**P**”. (b) If X and Y are terms, then $\langle X, Y \rangle$ is a term, too. (c) If X and Y are terms, then $X = Y$ and $X \in Y$ are formulas. (d) If φ and ψ are formulas, so are $\neg\varphi$ and $[\varphi \circ \psi]$ where \circ is one of the signs $\wedge, \vee, \rightarrow$, or \leftrightarrow . (e) If φ is a formula and v a variable, then $\exists v.\varphi$ and $\forall v.\varphi$ are formulas and $\imath v.\varphi$ and $\{v \mid \varphi\}$ terms. Terms denote either individuals or classes. In the following we shall use the letters “ a ” and “ b ” (possibly with subscripts) for terms of the first kind. For terms denoting classes of individuals we shall use the letters “ A ” and “ B ”; finally, the letter “ R ” is reserved for classes of tuples of individuals.

The logic for the connectives and quantifiers is classical with two exceptions. First, in order to exclude certain trivial cases, LC requires that there are at least two individuals ($\exists u_1 u_1. u_1 \neq u_2$) whereas one postulates in the semantics of standard predicate logic only that the universe of discourse is not empty. The second difference concerns the rule SUB of substitution of free variables by terms. The presence of class terms in LC make it necessary to restrict this rule in order to protect the system against antinomies. In order to formulate the rule, we have first to define the notion of the domain D^s of sort s .

$$\text{Def 1: } \begin{array}{ll} \text{(a)} & D^s \stackrel{\text{def}}{=} \{v^s \mid v^s = v^s\} \\ \text{(b)} & D \stackrel{\text{def}}{=} D^u \end{array}$$

Let now in the following formulation of the rule SUB $\varphi(v^s)$ be a formula with the free variable v^s and X a term which does not contain any free variable which is bound by a quantifier of $\varphi(x)$ in whose scope v^s occurs as a free variable, then we denote by “ $\varphi_{v^s}^X$ ” the result of substituting each free occurrence of v^s in $\varphi(v^s)$ by X . The rule SUB, then, reads as follows.

SUB From $X \in D^s$ and $\varphi(v^s)$ one may infer $\varphi_{v^s}^X$.

The reason for the additional premise becomes obvious as soon as we consider the class theory of LC. It consists of three principles: the principle of extensionality and two abstraction principles.

$$\begin{array}{ll} \text{LC 1: } & \text{(Ext)} \quad \forall u. [u \in \{v \mid \varphi(v)\} \leftrightarrow u \in \{w \mid \psi(w)\}] \rightarrow \\ & \quad \{v \mid \varphi(v)\} = \{w \mid \psi(w)\} \\ & \text{(Abs}_1\text{)} \quad v \in \{v \mid \varphi\} \leftrightarrow \varphi \\ & \text{(Abs}_2\text{)} \quad X \in \{v^s \mid \varphi\} \rightarrow X \in D^s \end{array}$$

⁷Mereology is one of the two logical theories which the Polish logician Stanisław Leśniewski proposed as frameworks for the explication of the traditional notion of a class. Leśniewski discerned two different meanings within that notion, namely that of a “distributive” and that of a “collective” class. Collective classes are treated in mereology, i.e. the theory of the part-of-relationship, whereas distributive classes are the topic of what he called “ontology”, the theory of the is-a-relationship. The relationship between common set theory and mereology has been investigated in, e.g., [20] and [21, esp. chs. 5 and 7]. Such a comparison, however, is a delicate issue since Leśniewski based mereology upon his ontology which is a more powerful logic than elementary predicate logic; cf. [22] for a detailed discussion. In the present article, too, (atomistic) mereology is transplanted into a non-Leśniewskian framework.

⁸It should be pointed out here that this is a quite comprehensive (and non-standard) concept of a region. Tarski [23, p. 24] suggests that the “solids” of the geometry envisaged by Leśniewski and the “events” of Whitehead’s space-time are “intuitive correlates of open (or closed) regular sets”. This (or something like this) seems to be true also for the common systems of mereotopology. The set of points corresponding to a region in the sense explained above in the main text, however, does not need to be regular.

The formation rules allow to build such a term as, e.g., $\{u | u \notin u\}$. It is nevertheless not possible to derive Russell's antinomy by means of (Abs_1) since (Sub) licenses the substitution of $\{u | u \notin u\}$ for the free variable v in $(\text{Abs})_1$ only under the proviso that the class $\{u | u \notin u\}$ can be proven to be an individual of sort s . Russell's antinomy shows that this cannot be the case for any s .

The theory of identity contained in LC is quite standard. There are two axioms requiring that identity is reflexive and euclidean. A further axiom finally postulates that identical individuals belong precisely to the same classes.

$$\begin{aligned} \text{LC 2: } & (\text{Id})_1 \quad X = X \\ & (\text{Id})_2 \quad X = Z \wedge Y = Z \rightarrow X = Y \\ & (\text{Id})_3 \quad X_1 = X_2 \wedge Y_1 = Y_2 \rightarrow [X_1 \in Y_1 \leftrightarrow X_2 \in Y_2] \end{aligned}$$

As is common LC construes relations as classes of pairs. There are two axioms for pairs. The first one states the usual criterion of identity for pairs: they are the same iff their components are. The second axiom postulates that pairs of individuals are individuals again.

$$\begin{aligned} \text{LC 3: } & (\text{Pr})_1 \quad \langle X_1, Y_1 \rangle = \langle X_2, Y_2 \rangle \leftrightarrow [X_1 = X_2 \wedge Y_1 = Y_2] \\ & (\text{Pr})_1 \quad X, Y \in \mathbb{D} \rightarrow \langle X, Y \rangle \in \mathbb{D} \end{aligned}$$

The use of the description operator ι is regulated by three axioms. The first requires that $\iota v.\varphi$ is the individual v if there is exactly one φ and $\varphi(v)$. If there is no unique individual with property φ , then the definite description $\iota v.\varphi(x)$ denotes a special “ersatz” individual \perp called “the joker”. The joker may be defined by $\perp \stackrel{\text{def}}{=} \iota u_0.u_0 \neq u_0$. It does not belong to the universe of discourse \mathbb{D} but is a virtual object.

$$\begin{aligned} \text{LC 4: } & (\text{Ds})_1 \quad \varphi(v) \wedge \exists^1 v.\varphi(x) \rightarrow \iota v.\varphi(v) = v \\ & (\text{Ds})_2 \quad \neg \exists^1 v.\varphi(v) \rightarrow \iota v.\varphi(x) = \perp \\ & (\text{Ds})_3 \quad \perp \notin \mathbb{D} \end{aligned}$$

Though there are no comprehension principles in LC, an elementary theory of classes and relations can be developed which provides most of the means of expressions which became common since the days of Cantor. X is a class if it is identical with its *class part*, i.e., with the class of individuals which are elements of X .

$$\text{Def 2: } \text{Cls}(X) \stackrel{\text{def}}{\iff} X = \{u | u \in X\}$$

We note that the referents of abstraction terms are always classes; cf. [13, p. 235].

$$\text{The 1: } \text{Cls}(\{v | \varphi\})$$

Inclusion relates subclasses to superclasses.

$$\text{Def 3: } A \subseteq B \stackrel{\text{def}}{\iff} \text{Cls}(A) \wedge \text{Cls}(B) \wedge \forall u.[u \in A \rightarrow u \in B]$$

The Boolean operations may be defined in the standard way. Relations are classes of n -tuples.

$$\begin{aligned} \text{Def 4: } & \text{Rel}^n(R) \stackrel{\text{def}}{\iff} \text{Cls}(R) \wedge \\ & \forall u \in R. \exists u_1 u_2 \dots u_n. u = \langle u_1, u_2, \dots, u_n \rangle \end{aligned}$$

As usual, the inverse of a relation results from that relation by inverting the order of its pairs.

$$\text{Def 5: } R^{-1} \stackrel{\text{def}}{=} \{\langle u_1, u_2 \rangle | \langle u_2, u_1 \rangle \in R\}$$

The two definitions below introduce abbreviations used in the following.⁹

$$\begin{aligned} \text{Def 6: } & (a) \quad R^>a \stackrel{\text{def}}{=} \{u | \langle u, a \rangle \in R\} \\ & (b) \quad R^<a \stackrel{\text{def}}{=} \{u | \langle a, u \rangle \in R\} \end{aligned}$$

Functions are defined as special relations fulfilling a uniqueness condition:

$$\begin{aligned} \text{Def 7: } & \text{Fct}^n(R) \stackrel{\text{def}}{\iff} \text{Rel}^{n+1}(R) \wedge \\ & \forall u_1 \dots u_{n+2}. [\langle u_1, \dots, u_n, u_{n+1} \rangle, \\ & \langle u_1, \dots, u_n, u_{n+2} \rangle \in R \rightarrow u_{n+1} = u_{n+2}] \end{aligned}$$

Functions will often be defined by specifying how to determine the value a for given arguments u_1, u_2, \dots, u_n . Given a certain (n -place) function f , the term “ $f(a_1, a_2, \dots, a_n)$ ” will denote the value of f for the arguments a_1, a_2, \dots, a_n (if it exists).

$$\begin{aligned} \text{Def 8: } & (a) \quad \lambda u_1 u_2 \dots u_n. a \stackrel{\text{def}}{=} \{\langle u_1, u_2, \dots, u_n, u \rangle | u = a\} \\ & (b) \quad f(a_1, a_2, \dots, a_n) \stackrel{\text{def}}{=} \iota u. \langle a_1, a_2, \dots, a_n, u \rangle \in f \end{aligned}$$

Functions may be partial. In mereology, for instance, the *product* $u_1 \cdot u_2$ of two individuals is the largest individual (*modulo* the part-of-relation) which is a common part of both u_1 and u_2 . If in a formal system of mereology the product operation \cdot is not taken as primitive, it will be defined by some function involving a definite description; cf., e.g., [28, p. 43]. That description term will be improper if the items denoted by the “factor” terms do not overlap. Since no product exists in this case, the product operation is partial. In LC the product of non-overlapping regions u_1 and u_2 equals the joker: $u_1 \cdot u_2 = \perp$.

B. Atomistic Mereology

Whitehead [29]–[31] motivates his use of mereological concepts as a foundation for his space-time-geometry by the desire for a conceptual framework which directly relates this science to spatial reality rather than starting from abstractions such as, e.g., extensionless points and breadthless lines. Using his “method of extensive abstraction”, he constructed such entities from extended regions. The critique of such notions as that of a point and that of a line, however, is much older and in fact nearly as old as the science of geometry itself.¹⁰ In the 19th century, Lobachevski and Bolyai did not only replace Euclid's Fifth Axiom (on the unique existence of parallels) by other assertions but also suggested to take the notion of a

⁹The notation introduced in Def 6 is often used to render formulas more easily readable. E.g., “ $u_1 \in P^>u_2$ ” can be read from left to right as “ u_1 is (\in) a part of ($P^>$) u_2 ”. The same is said by “ $\langle u_1, u_2 \rangle \in P$ ”, which however requires the reader to apply a “forth and back” procedure when decoding the formula. — “ $u_1 \in P^<u_2$ ” may be read as “ u_1 extends / is an extender of u_2 ”.

¹⁰Cf., for instance, Aristotle's remark in his *Metaphysics*, [32, p. 36, 992^a 20] that Plato “fought against [the kind of points] as being a geometric dogma” and Proclus Lycaeus warning — in his commentary on the first book of Euclid's *Element* — not to follow the Stoics who suppose that such limiting elements like points “exist merely as the product of reflection”; [33, p. 71].

rigid, three-dimensional body as the conceptual starting point for geometry.¹¹ In a similar vein, Whitehead [29]–[31], relying on a certain analysis of the role of abstractions in science, developed a theory of events which bears some similarities to Leśniewski's mereology, which Tarski [23] combined with ideas of the Italian mathematician Mario Pieri in order to formulate his geometry of solid bodies.¹² This line of research has been continued by the work of Gruszczyński and Pietruszczak [26]. Whitehead's approach, especially as modified in his book from 1929, has been continued in "mereotopology"—cf., e.g., [28] and [27]—and in work on the Region-Connection-Calculus (RCC); cf. [38] and the literature cited there.

Hahmann *et al.* [27, p. 1424] formulate the objections against points in a concise way: "Points are somewhat tricky to define and are far from intuitive in real-world applications." It is certainly true that the definition of points as equivalence classes of converging sequences of regions (as suggested by Whitehead and others) is "tricky".¹³ However, if points are admitted from the outset as special ("extensionless") regions, it is rather easy to single them out by a definition. Actually, we find an adequate explanation already as the very first definition in the first book of Euclid's *Elements*: "A point is that which has no part"; [2, Book I, p. 153]. As is evident from this definition, Euclid obviously is thinking of the proper (irreflexive) part-of-relationship when he is talking about parts. Allowing, as is usual in mereology, also for improper parts, we can reformulate Euclid's definition as follows: "A point is a minimum of the part-of-relation."

Euclid's definition testifies that the notion of a point nicely fits into a mereological framework. Aside from this formal issue, perceptual psychology does not seem to support the sceptical attitude towards points held up by many supporters of "common sense geometry". Experimental studies of visual space simply accept the existence of points when they approximate these geometric items by "small point sources of light of low illumination intensity, displayed in darkened room;" [46, p. 238]. Points seem also to be accepted in phenomenological

and gestalt-theoretic approaches to psychology. In a series of classical experiments Edgar Rubin [47, §§14–16] showed that points (as well as other regions lacking extension in one or more dimension) are really perceived: "As there are breadthless lines, there are extensionless points". Furthermore, Otto Selz [48, p. 40] argued that points essentially belong to our conceptual frame used in the apprehension of space: "the pure location in space is postulated by structural laws in the same way as the infinity of the straight line and [...] it is of relatively minor importance whether the empirical *Minimum Visibile*, i.e., the point gestalt, is to be regarded as a pure locational phenomenon or rather as a tiny round area like object". We hence conclude that points, though they are perhaps no "real constituents" of physical space, do have perceptual reality and exist in conceptualized space. This is all which is of importance in the present context.

As the mereological foundation of our system of geometry we adopt the system of atomistic mereology developed by Tarski; cf. [49]. Tarski formulated his system within the simple theory of types. Instead we use the class logic LC sketched in the previous subsection. The only undefined notion in Tarski's system is the relation P of parthood¹⁴ of which it is postulated that it is transitive. In LC this correspond to the following two axioms.

MER 1: $\text{Rel}^2(P)$

MER 2: $\langle u_1, u_2 \rangle, \langle u_2, u_3 \rangle \in P \rightarrow \langle u_1, u_3 \rangle \in P$

We say that two individuals (regions) overlap if they share a common part.¹⁵

Def 9: $O \stackrel{\text{def}}{=} \{ \langle u_1, u_2 \rangle \mid P^{>}u_1 \cap P^{>}u_2 \neq \emptyset \}$

The formulation of the next axioms requires the following definition.

Def 10: $\Sigma(a, A) \stackrel{\text{def}}{\iff} A \subseteq P^{>}a \wedge \forall u_1 \in P^{>}a. \exists u_2 \in A. \langle u_1, u_2 \rangle \in O$

The formula " $\Sigma(a, A)$ " says that a is the mereological sum of the individuals in A . This means that every element of A is a part of a and that conversely every part of a overlaps some element of A . The mereological sum of a singleton class is the unique member of that class; and non-empty classes always have a sum.

MER 3: $\Sigma(u_1, \{u_2\}) \rightarrow u_1 = u_2$

MER 4: $A \neq \emptyset \rightarrow \exists u. \Sigma(u, A)$

From the axioms stated until now it can be proven¹⁶ that P is a partial order of the elements of \mathbb{D} , i.e., that the part relation, besides being transitive, is reflexive and antisymmetric. Furthermore, *MER 4* may be strengthened by asserting the uniqueness of the mereological sum.

¹⁴In [49] Tarski augments his system of pure mereology by other non-mereological systems in order to make it suitable as a basis for axiomatic biology.

¹⁵The notion of overlap is not used by Tarski. We introduce it here in order to make our presentation more similar to standard expositions of mereology; cf., e.g., [28].

¹⁶For the proofs of the mereological theorems the reader is referred to Tarski's article [49].

¹¹Their endeavors are described in Richard Strohal's investigations of the relationships between "pure geometry" and intuition; cf. [34, pp. 20–33]. It deserves to be mentioned that Lobacevski considered the relationship of connection (between "solids") as the most basic concept of geometry thus anticipating the modern line of research which starts with the work of de Laguna [35] and leads up via Whitehead's reformulation of his earlier work in [31] to the Region Connection Calculus of Randell, Cui, and Cohn [17].

¹²In his lecture notes [36], Leśniewski compares his mereology with Whitehead's theory of events. In those notes, Leśniewski mentions that it was Tarski in 1926 who made him aware of Whitehead's work; cf. [36, p. 171]. — Pieri's idea employed by Tarski [23] in his mereological system of geometry is that this discipline can be developed starting from the notions of point and sphere as the only undefined concepts. Pieri's memoir presenting this idea has been re-published in an English translation by Marchisotto and Smith [37, pp. 157–288]. This book contains also a chapter on Pieri's impact on Tarski's geometric work; cf. [37, ch. 6]. That the notion of sphere is sufficient as a basic concept of geometry has been noted already by Grassmann in his 1847 memoir on Leibniz' geometric analysis; cf. [7, p. 328].

¹³The issue of "region-based" vs. "point-based" geometry is treated in quite a few articles on mereotopology; cf. [39] and [40], who both provide surveys of the classical approaches to this topic by Whitehead [29]–[31], De Laguna [35], Menger [41], Grzegorzczak [24], and Clarke [25]. More recent contributions include [42], [43], [27], [44], and [45].

- The 2:* (a) $\langle u, u \rangle \in P$
 (b) $\langle u_1, u_2 \rangle, \langle u_2, u_1 \rangle \in P \rightarrow u_1 = u_2$
 (c) $A \neq \emptyset \rightarrow \exists^1 u. \Sigma(u, A)$

The 2-(c) justifies the following definitions introducing the notion of a supremum or sum of a class of individuals.

- Def 11:* (a) $\sup(A) \stackrel{\text{def}}{=} \lambda u. \Sigma(u, A)$
 (b) $\sup(v^s \mid \varphi) \stackrel{\text{def}}{=} \sup(\{v^s \mid \varphi\})$
 (c) $\sup(a_1, a_2, \dots, a_m) \stackrel{\text{def}}{=} \sup(\{a_1, \dots, a_m\})$
 (d) $+ \stackrel{\text{def}}{=} \lambda u_1 u_2. \sup(u_1, u_2)$

Of course, we shall always write “ $a + b$ ” instead of “ $+(a, b)$ ”. It can be proven that the thus defined notion of a supremum of A has indeed the properties normally required: namely, that it is the “smallest” individual “bigger” than all the elements of A , cf. *The 3*.

- The 3:* $u = \sup(A) \rightarrow A \subseteq P^>u \wedge \forall u_1. [A \subseteq P^>u_1 \rightarrow u \in P^>u_1]$

It is provable in LC that \mathbb{D} is non-empty; hence by *The 2*-(c) there exists the sum of all individuals. Following [49, p. 162], we shall call it \mathbf{w} (which Tarski transliterates as “world”). It is the entire space.

- Def 12:* $\mathbf{w} \stackrel{\text{def}}{=} \sup(\mathbb{D})$

The space is an individual, hence it exists, and everything, i.e., every region, is a part of it; cf. *The 4*.¹⁷

- The 4:* (a) $\mathbf{w} \in \mathbb{D}$
 (b) $u \in P^>\mathbf{w}$

Corresponding to the notion of the mereological sum of a class of individuals there is the notion of a product. This is not defined by Tarski; but *Def 13* suggests itself by its analogy to the case of the sum.

- Def 13:* (a) $\Pi(a, A) \stackrel{\text{def}}{\iff} \Sigma(a, \{u \mid \forall u_1 \in A. \langle u, u_1 \rangle \in P\})$
 (b) $\inf(A) \stackrel{\text{def}}{=} \lambda u. \Pi(u, A)$
 (c) $\inf(v^s \mid \varphi) \stackrel{\text{def}}{=} \inf(\{v^s \mid \varphi\})$
 (d) $\inf(a_1, a_2, \dots, a_m) \stackrel{\text{def}}{=} \inf(\{a_1, a_2, \dots, a_m\})$
 (e) $\cdot \stackrel{\text{def}}{=} \lambda u_1 u_2. \inf(u_1, u_2)$

Again we use infix notation “ $a_1 \cdot a_2$ ” instead of “ $\cdot(a_1, a_2)$ ”. There is an important difference between the notion of a sum and that of a product: whereas only non-empty classes have a sum, also the empty class has a product. If $A = \emptyset$, then $\{u \mid \forall u_1 \in A. \langle u, u_1 \rangle \in P\} = \mathbb{D}$ and hence $\Pi(A, \mathbf{w})$ according to *Def 12*. This, however, does not mean that the infimum always exists. If A is a class of non-overlapping individuals, i.e., of individuals which have no common parts, then the class $\{u \mid \forall u_1 \in A. \langle u, u_1 \rangle \in P\}$ will be empty and will hence not have a supremum. In this case, therefore, A will not have an infimum.

¹⁷Theorems like *The 4*-(a) are of special importance for our formal framework since the rule of substitution of LC is, as has been explained in section II-A above, restricted in such a way that the substitution of a term X for a variable of sort s requires a proof of $X \in D^s$.

Corresponding to *The 3* we have the following theorem for the infimum.

- The 5:* $u = \inf(A) \rightarrow A \subseteq P^<u \wedge \forall u_1. [A \subseteq P^<u_1 \rightarrow u_1 \in P^<u]$

According to *The 4*, \mathbf{w} is the unique region which is maximal with respect to the P -relation. Now after we have decided to adopt points as the minima of that relation, it is useful also to adopt a special sort \mathbf{p} for points. Hence $D^{\mathbf{p}}$ (cf. *Def 1*-(a)) is the class of all points which thus does not need a special definition. However, in order to catch the identification of points with P -minima, we have to accept a special axiom which corresponds to Tarski’s Definition of points; cf. [49, p. 163].

- MER 5:* $D^{\mathbf{p}} = \{u \mid P^>u \subseteq \{u\}\}$

MER 5 has still to be supplemented by Tarski’s postulate that each individual has at least one punctual part.

- MER 6:* $P^>u \cap D^{\mathbf{p}} \neq \emptyset$

MER 6 is sufficient to show that each individual is the sum of its points; Tarski’s proof for this can be transferred to the present system.

- The 6:* $u = \sup(p \mid p \in P^>u)$

However, within the framework of LC this does not mean that talk about regions can be dismissed in favour of talk about point classes since within LC (unlike as in Tarski’s type-theoretic framework) we cannot quantify over point classes though quantification over regions is possible.

III. INTERVAL SPACES AND CONVEXITIES

In the previous sections we have laid the logical and mereological foundations for the system of geometry which will be presented in a stepwise manner in this and the following two sections. In the first part of the present section, we do not extend the foundational framework provided by any further axioms but define some concepts of central importance for our system of geometry. Then we point out some simple consequences which can be derived from the definitions given only by means of logic and mereology. In the second part of the present section we then state the first axioms of a geometric character.

A. Pregeometry

By a convex region we understand a region in which every pair of points is connected by a linear segment completely belonging to that region. A triangle and a circle are examples of convex regions whereas the bean shaped region of Fig. 1 is not. The variables of sort \mathbf{c} vary over the elements of the domain $D^{\mathbf{c}}$ (cf. Def. 1) which is the class of all convex regions. *Def 14* introduces the central notion of the *convex hull* of a region: the function $[]$ assigns to each region u its convex hull $[](u)$. We write “ $[u]$ ” instead of “ $[](u)$ ” in order to comply with ordinary notation. The convex hull $[u]$ is the infimum of all convex regions containing u as a part.

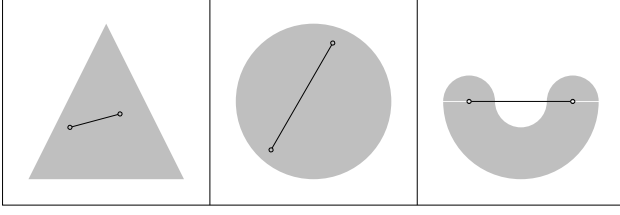


Fig. 1. Two convex regions and a non-convex one

- Def 14: (a) $[] \stackrel{\text{def}}{=} \lambda u. \inf(c \mid u \in P^>c)$
 (b) $[p_1, p_2, \dots, p_n] \stackrel{\text{def}}{=} [\sup(p_1, p_2, \dots, p_n)]$
 (c) $p_1 p_2 \stackrel{\text{def}}{=} [p_1, p_2]$

Def 14(b) defines the *polytope* — or, more precisely, the *n*-tope — spanned by p_1, p_2, \dots, p_n as the convex hull of the sum of those points. The *segment* $p_1 p_2$, then, between p_1 and p_2 is just the 2-tope spanned by these two points, cf. Def 14(c). These definitions correspond to those given in point set based convex geometry; cf., e.g., [1, pp. 3, 5]. The convex hull $[u]$ of a region u always exists (is an individual) and contains u as a part.

- The 7: (a) $[u] \in \mathbb{D}$
 (b) $u \in P^>[u]$

Proof: Let $A := \{u \mid \forall c. [\langle u, c \rangle \in P \rightarrow \langle u_1, c \rangle \in P]\}$. If there are no convex regions containing u , then $A = \mathbb{D}$ and $[u] = \mathbf{w} \in \mathbb{D}$; and (b) holds by The 4. If, however, there are convex regions containing u , then $u \in A \neq \emptyset$ and hence $[u] \in \mathbb{D}$ by The 2-(c). The assertion (b), then, follows by The 5. ■

From The 7 we have immediately The 8-(a); the second claim of that theorem is a direct consequence of the definition of the product and the hull operation.

- The 8: (a) $p_1, p_2 \in P^>p_1 p_2$
 (b) $p_1 p_2 = p_2 p_1$

The 8 states (*modulo* the replacement of set theoretical notions by mereological ones) that the class of points and that of segments together with the segment operation constitutes an *interval space*; cf. [1, chap. 1, Sec. 4].¹⁸ The 8-(a) corresponds to the so-called *extensive law*, Theorem 8-(b) to the *symmetry law* of interval spaces. Interval spaces in turn give rise to *convex structure* (also briefly called *convexities*). These are the structures exhibiting the basic facts about convex sets; cf. [1, p. 3]. A convexity is a family C of subsets of some point set X which fulfills the following three closure conditions:

- (C-1) $\emptyset, X \in C$;
 (C-2) for $\mathcal{D} \subseteq C$ is $\bigcap \mathcal{D} \in C$;
 (C-3) if for $A, B \in \mathcal{D} \subseteq C$ it always holds true that $A \subseteq B$ or $B \subseteq A$, then $\bigcup \mathcal{D} \in C$.

¹⁸On the background of set theory, an interval space $I = \langle X, I \rangle$ is defined to be a pair consisting of a set X of points and an operation $I : X \times X \rightarrow 2^X$ such that for $p, q \in X$ it holds true that $p, q \in I(p, q)$ and $I(p, q) = I(q, p)$; cf. [1, p. 71].

Using the segment operation, we define in our mereological context the special class C_v of regions in the following way.

$$C_v \stackrel{\text{def}}{=} \{u \mid \forall p_1, p_2 \in P^>u. p_1 p_2 \in P^>u\}$$

It is not too difficult to show that C_v fulfills mereological analogues to (C-1), (C-2), and (C-3).

- The 9: (C-1)' $\mathbf{w} \in C_v$
 (C-2)' $\exists u. A \subseteq P^<u \wedge A \subseteq C_v \rightarrow \inf(A) \in C_v$
 (C-3)' $\emptyset \neq A \subseteq C_v \wedge \forall u_1, u_2 \in A. \langle u_1, u_2 \rangle \in P \cup P^{-1} \rightarrow \sup(A) \in C_v$

Proof: (C-1)' is immediate from The 4-(b). — (C-2)'. The first conjunct of the hypothesis ensures that $u_1 := \inf(A) \in \mathbb{D}$. It remains to be shown that each segment $p_1 p_2$ where $p_1, p_2 \in P^>u_1$ is itself a part of u_1 . From $p_1, p_2 \in P^>u_1$ it follows by The 5 that for each $u_2 \in A$ $p_1, p_2 \in P^>u_2$ and hence $p_1 p_2 \in P^>u_2$ since $A \subseteq C_v$. Thus $p_1 p_2 \in u_2$ for each $u_2 \in A$, hence $u_1 \in C_v$. — (C-3)'. Since $A \neq \emptyset$, again $u_1 := \sup(A) \in \mathbb{D}$. Suppose $p_1, p_2 \in P^>u_1$. According to Def 10, the two points share, respectively, a part with two individuals $u_2, u_3 \in A$. According to the second conjunct of the assumption $\langle u_2, u_3 \rangle \in P$ or, conversely, $\langle u_3, u_2 \rangle \in P$. Assume the first (the argument for the second is completely parallel). Then $p_1, p_2 \in P^>u_2$ and, since $u_2 \in A \subseteq C_v$, $p_1 p_2 \in P^>u_2$. But then $p_1 p_2 \in P^>u_1$, too. Hence $u_1 = \sup(A) \in C_v$. ■

A region u belongs to C_v if all the “2-topes”, i.e., segments, whose boundary points are from u lie within that very region. Of course, the definition of the class C_v is an exact formal counterpart of the intuitive explanation of the notion of a convex region provided at the beginning of this subsection. Therefore it cannot be included as a formal definition within our system since this would involve a circularity: segments are defined in terms of convex regions (by using variables of sort c), hence one cannot use segments in order to define convex regions. However, the class C_v should turn out to be identical with the domain D^c . Within pregeometry we can prove at least the inclusion of that domain in C_v ; cf. The 10. The converse inclusion will be postulated as an axiom in the next subsection.

The 10: $D^c \subseteq C_v$

Proof: This follows readily from The 5 and Def 14. ■

B. Convex Structure

Spelled out, the converse of The 10 amounts to the following principle.

$$\text{GEO 1: } \forall p_1, p_2 \in P^>u. p_1 p_2 \in P^>u \rightarrow u \in D^c$$

By GEO 1 we leave mereology and pregeometry and enter the realm of geometry proper. Therefore the label “GEO” is given to the new axiom rather than continuing using “MER” in order to mark principles. By The 10 we may strengthen GEO 1 to a biconditional.

$$\text{The 11: } u \in D^c \leftrightarrow \forall p_1, p_2 \in P^>u. p_1 p_2 \in P^>u$$

Furthermore, we may now replace “ C_v ” in The 10 by “ D^c ”.

- The 12:* (a) $w \in D^c$
 (b) $\exists u. A \subseteq P^<u \wedge A \subseteq D^c \rightarrow \inf(A) \in D^c$
 (c) $\emptyset \neq A \subseteq D^c \wedge$
 $\forall u_1, u_2 \in A. \langle u_1, u_2 \rangle \in P \cup P^{-1} \rightarrow$
 $\sup(A) \in D^c$

The 11 and *The 12* state that D^c is a (mereological) convex structure. *The 12*-(b) immediately implies that convex hulls are, as their name suggests, convex. A corollary of this is that segments, which are special convex hulls—namely convex hulls of regions consisting of at most two points—are convex.

- The 13:* (a) $[u] \in D^c$
 (b) $D^s \subseteq D^c$

The 13-(a) implies that the hull of a region's hull equals that hull and that the hull operation is monotonic.¹⁹

- The 14:* (a) $[[u]] = [u]$
 (b) $u_1 \in P^>u_2 \rightarrow [u_1] \in P^>[u_2]$
 (c) $u \in P^>c \rightarrow [u] \in P^>c$

Proof: By *The 7*-(b), *The 5*, and *The 13*. ■

The 7-(b) and *The 14*-(a), (b) state that the $[\]$ -operator is a hull-operator in the algebraic sense. As a special case of *The 14*-(b) we have that segments spanned by the points of some given segment are subsegments of that segment and that hence a point of a given segment dissects a subsegment of that segment.

- The 15:* (a) $p_1, p_2 \in P^>s \rightarrow p_1p_2 \in P^>s$
 (b) $p_3 \in P^>p_1p_2 \rightarrow p_1p_3 \in P^>p_1p_2$

The 15-(b) is called the *monotone law* in [1, p. 74]. — We did not require that the two arguments of the segment operator are distinct. If we ask for the segment $[pp]$ joining the point p to itself, a natural answer would be that in this case the segments shrinks down to the point p . Hence points are just segments without any extension. Points are also special regions: they are the minimal regions. But then, we have to admit that the only segment which is a part of a minimal region p is p itself and that therefore p is convex according to our intuitive explanation of convexity. That this is actually the case is postulated by a new axiom which states that the points are a *subsort* of the convex regions.

- GEO 2:* $D^p \subseteq D^c$

From *GEO 2* it is immediate that points are minimal segments. This is, for obvious reasons, called the *idempotent law* in the theory of interval spaces; cf. [1, p. 74].

- The 16:* $pp = p$

Proof: From *GEO 2* together with *The 5* and *The 7*. ■

The domain D^s is characterized by the following axiom.

- GEO 3:* $D^s = \{u \mid \exists p_1, p_2. u = p_1p_2\}$

From *The 16* and the new axiom *GEO 3* it follows that points are special segments, namely one-point-only segments.²⁰

- The 17:* $D^p \subseteq D^s$

Another consequence of *The 16* is that each region is the sum of its segmental parts.

- The 18:* $u = \sup(s \mid s \in P^>u)$

Proof: By *The 6* and *The 16* a region is already the sum of its punctual segments. The non-punctual elements of the class $\{s \mid s \in P^>u\}$ do not add anything more to the mereological sum of this class. ■

In the case of convex regions *The 18* can be given the following strengthened form.

- The 19:* $p_1 \in P^>c \rightarrow c = \sup(s \mid \exists p_2 \in P^>c. s = p_1p_2)$

Proof: From $p_1 \in P^>c$, it follows by *The 11*, that $p_1p_2 \in P^>c$ for each $p_2 \in P^>c$. Hence $\sup(s \mid \exists p_2 \in P^>c. s = p_1p_2) \in P^>c$. — It remains to be shown that also conversely $c \in P^>\sup(s \mid \exists p_2 \in P^>c. s = p_1p_2)$. Assume so that $p_3 \in P^>c$. It suffices to show that $p_3 \in P^>\sup(s \mid \exists p_2 \in P^>c. s = p_1p_2)$. But this follows readily from *The 16*. ■

To conclude the present subsection, we state a further axiom which strengthens the theorem just proven for a special kind of convex regions. Consider some point p and a convex region c . The region $[p + c]$ may be called the *cone* with *apex* p and *base* c ; cf. Fig. 2. Since the cone has been constructed as a hull, it is convex. By *The 19*, then, it equals the sum of all the segments starting from the apex and ending at some other point of the cone. The next axiom states that we do not really need to consider all segments of the kind described but rather can restrict ourselves to segments from the apex to the points of the base (as the points p_1, p_2 and p_3 in Fig. 2).

- GEO 4:* $[p_1 + c_1] = \sup(s_1 \mid \exists p_2 \in P^>c_1. s_1 = p_1p_2)$

This axiom is called *join-hull commutativity* since it postulates that the hull operation and the sum (“join”) operation may be interchanged; cf. [1, p. 39]. The reader should remember here that pp_1 actually is the convex hull $[p_1, p_2]$. To make thus the name of the principle more transparent, we could render it as $[\sup(p_1, c_1)] = \sup([p_1p_2] \mid p_2 \in P^>c_1)$.²¹

IV. STRAIGHTNESS AND ORDER

In the previous section we dealt with the relationship between segments and convex regions. Segments connect the points of a convex region without leaving that region. In the present section we shall consider two further important properties of segments. In the first subsection we shall set up two axioms which make explicit what it means for a segment to be “straight” rather than “bent”. Then we shall study the order of points in a segment.

²⁰Though *The 17* seems to be quite trivial, its formal proof requires some care as regards the handling of the sorts.

²¹Where “ $\sup([p_1p_2] \mid \varphi)$ ” abbreviates “ $\sup(s_1 \mid s_1 = p_1p_2 \wedge \varphi)$ ”.

¹⁹Of the five axioms stated by Randell et. al. [17, p. 5] for their operator *conv*, the first one corresponds to *The 7*-(b) and the second to *The 14*-(a). The third axiom follows easily from *The 14*-(b) (in combination with the (a)-clause of that theorem). The two remaining axioms which relate the concept of a convex hull to the relation *O* (of overlap) and its complement, are true in the “intended model” of the present theory, too. A proof of them, however, is by no means obvious.

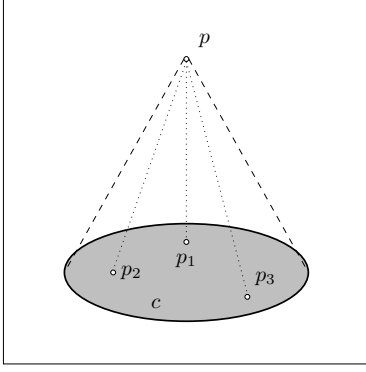


Fig. 2. Join-hull commutativity.

A. Straightness

Segments are convex, as we have seen, and so are the two first sample regions (the triangle and the circle) displayed in Fig. 1. Segments are one-dimensional and thus differ from triangles and circles which are two-dimensional. There is yet another property which sets segments apart from circles. A circle (by its circumference) involves curvature whereas a segment is *straight*. In the present subsection we set up two axiomatic principles which make explicit what it means to be straight. The first of these two principles is known as *decomposability*; cf. [1, p. 143]. In our framework it may be rendered as follows.

$$\text{GEO 5: } p_2 \in P^>p_1p_3 \rightarrow p_1p_3 = p_1p_2 + p_2p_3 \wedge p_2 = p_1p_2 \cdot p_2p_3$$

A point of a segment dissects the whole segment into two component segments which overlap precisely in the dissecting point; cf. the left diagram of Fig. 3. Hence a curved line with a loop such as that displayed by the right diagram of Fig. 3 cannot be a segment since there is a point on that line which dissects it into three parts.

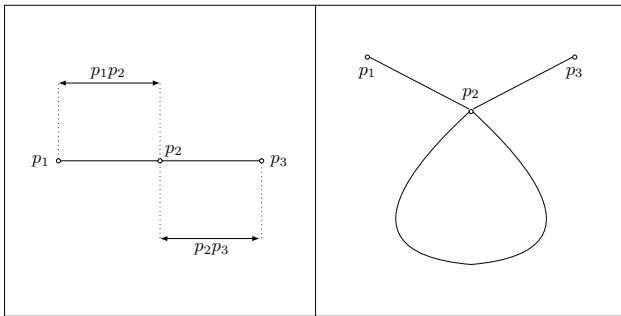


Fig. 3. Decomposition of a segment into two segmental components

As an immediate consequence of decomposability we have:

$$\text{The 20: } p_2 \in P^>p_1p_3 \wedge p_3 \in P^>p_1p_2 \rightarrow p_2 = p_3$$

Proof: By decomposability $p_1p_3 = p_1p_2 + p_2p_3$ with $p_2 = \inf(p_1p_2, p_2p_3)$. But since $p_3 \in P^>p_1p_2 \cap P^>p_2p_3$, $p_2 = p_3$. ■

The second postulate which explains what it means for a line to be straight is known by the name of this property, i.e., *straightness*; [1, p. 143].

$$\text{GEO 6: } \exists p_1p_2.[p_1 \neq p_2 \wedge p_1, p_2 \in P^>s_1 \cap P^>s_2] \rightarrow s_1 + s_2 \in D^s$$

The sum of two segments sharing two points cannot result in a curved line because in that case at least one of the items combined would have already been bent; cf. Fig. 4.

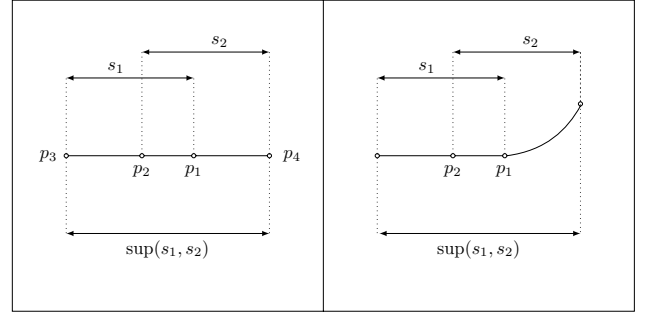


Fig. 4. The combination of two segments results in a straight segment again

The 5 and The 6 imply the *ramification* principle of The 21; cf. [1, p. 143] which says that two segments which have one boundary point in common but differ with respect to their second boundary will branch away from each other at the common point; cf. the left hand side of Fig. 5. The indirect proof of the ramification principle provided by [1, p. 144] within a set-theoretic framework can be directly transferred to our mereological system.

$$\text{The 21: } p_3 \notin P^>p_1p_2 \wedge p_2 \notin P^>p_1p_3 \rightarrow p_1 = p_1p_2 \cdot p_1p_3$$

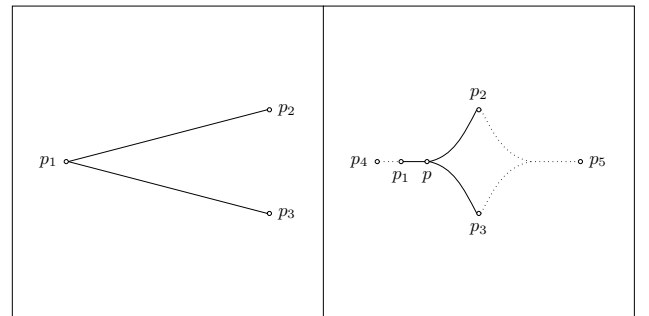


Fig. 5. The Ramification Property.

B. Order and Lines

Normally an order relation between points belongs to the undefined concepts of standard axiomatic systems of geometry; cf., e.g., [12, §3], [50, pp. 11–13]. In our framework such a relation may be defined.

$$\begin{aligned} \text{Def 15: } & \text{(a) } B \stackrel{\text{def}}{=} \{ \langle p_1, p_2, p_3 \rangle \mid p_2 \in P^>p_1p_3 \} \\ & \text{(b) } p_2 B p_1p_3 \stackrel{\text{def}}{\iff} \langle p_1, p_2, p_3 \rangle \in B \end{aligned}$$

Decomposability implies that of three points at least one lies between the two others.

The 22: $p_1, p_2, p_3 \in P^>s \rightarrow$
 $p_1Bp_2p_3 \vee p_2Bp_1p_3 \vee p_3Bp_1p_2$

Proof: If two of the three points mentioned in the assumption of the theorem are identical (say, e.g., $p_1 = p_2$), then the assertion surely holds true (since then, in the example case, $p_1Bp_2p_3$). Hence we assume that all the points differ from each other. Let p_4 and p_5 be the endpoints of the segment s ; hence $s = p_4p_5$. By *GEO 5* $s = p_4p_1 + p_1p_5$ with $p_1 = p_4p_1 \cdot p_1p_5$. Assume that p_2 and p_3 belong to different component segments of s , e.g.: $p_2Bp_4p_1$ and $p_3Bp_1p_5$ (the converse distribution is treated in analogous way). Two further applications of *The 5* yield $p_4p_1 = p_4p_2 + p_2p_1$ and $p_1p_5 = p_1p_3 + p_3p_5$ where p_2 and p_3 are, respectively, the only common points of the component segments. Hence both $p_1 \notin P^>p_4p_2$ and $p_1 \notin P^>p_3p_5$. Again by *GEO 5* we have $s = p_4p_2 + p_2p_5$ with $p_2 = p_4p_2 \cdot p_2p_5$. From the latter and $p_1 \notin P^>p_4p_2$ we conclude $p_1Bp_2p_5$ and, since $p_3Bp_1p_5$, $p_3Bp_2p_5$, too. A further application of *The 5* yields $p_2p_5 = p_2p_3 + p_3p_5$ with $p_3 = p_2p_3 \cdot p_3p_5$. We already know that $p_1 \notin P^>p_3p_5$ and thus infer $p_1Bp_2p_3$. — Now assume that p_2, p_3 belong to the same component segment of our first division of s into p_4p_1 and p_1p_5 . Assume $p_2, p_3 \in P^>p_4p_1$; the remaining possibility is treated in an analogous way. *The 5* yields that $p_4p_1 = p_4p_2 + p_2p_1$ with $p_2 = p_4p_2 \cdot p_2p_1$. Hence either $p_3Bp_1p_2$ and nothing remains to prove, or $p_3 \in P^>p_4p_2$. In the latter case we know by *The 20* that $p_2 \notin P^>p_4p_3$. But by a final application of *The 5* we have $p_4p_1 = p_4p_3 + p_3p_1$ with $p_3 = p_4p_3 \cdot p_3p_1$; therefore $p_2Bp_1p_3$. ■

Def 15 does not require that the two boundary points delimiting the position of the third point differ. If they do not, the point in between them is identical to them.

The 23: $p_1Bp_2p_2 \rightarrow p_1 = p_2$

This is immediate from the idempotent law for segments *The 16*. *The 23* is an axiom of Tarski's system of Euclidean geometry presented and investigated in [50]. There it is called the *identity axiom for the betweenness relation*. Pasch [51], who is celebrated for his analysis of the order relation, postulated that points connected by the B-relation differ from each other, and Hilbert [12] followed him in this. The B-relation assumed here may be easily modified in the way suggested by the Pasch-Hilbert-view.

Def 16: (a) $B^+ \stackrel{\text{def}}{=} B \cap \{\langle p_1, p_2, p_3 \rangle \mid \bigwedge_{1 \leq i < j \leq 3} p_i \neq p_j\}$
 (b) $p_2B^+p_1p_3 \stackrel{\text{def}}{\iff} \langle p_1, p_2, p_3 \rangle \in B^+$

For the concept of a segment and for betweenness-relation, which are both basic in his axiomatisation of geometry, Pasch [51, §1] postulates nine axioms. In the present framework they can be formulated as shown in Tab. I. Of these axioms, I. and IV. are just a special cases of *The 7* and the “monotone law” *Theorem 14*-(b), respectively. Furthermore, V. easily follows from *GEO 5*. VI. says that a segment p_1p_2 is always

extendable beyond its boundary point p_2 . We adopt it as a basic principle of our system.

GEO 7: $\exists p_3. p_2B^+p_1p_3$

IX. is a *dimensionality axiom*. Since we wish to remain neutral here with respect to dimensionality, we do not accept this axiom.

I.	$\exists s. s = p_1p_2$
II.	$\exists p_2. p_2B^+p_1p_3$
III.	$p_2B^+p_1p_3 \rightarrow \neg p_1B^+p_2p_3$
IV.	$p_2B^+p_1p_3 \rightarrow p_1p_2 \in P^>p_1p_3$
V.	$p_2B^+p_1p_3 \wedge p_4 \notin P^>p_1p_2 \cup P^>p_2p_3 \rightarrow p_4 \notin Pp_1p_3$
VI.	$\exists p_3. p_2B^+p_1p_3$
VII.	$p_2B^+p_1p_3 \wedge p_2B^+p_1p_4 \rightarrow p_3B^+p_1p_4 \vee p_4B^+p_1p_3$
VIII.	$p_2B^+p_1p_3 \wedge p_1B^+p_2p_4 \rightarrow p_1B^+p_3p_4$
IX.	$\exists p_3. [\neg p_1B^+p_2p_3 \wedge \neg p_2B^+p_1p_3 \wedge \neg p_3B^+p_1p_2]$

TABLE I

PASCH'S AXIOMS FOR SEGMENTS AND BETWEENNESS

Pasch's axiom II. is accepted here in the slightly modified but equivalent form *GEO 8*.

GEO 8: $p_1 \neq p_2 \rightarrow p_1p_2 \neq p_1 + p_2$

The 8 requires each non-punctual segment to contain at least two points, namely its boundaries. *GEO 8* excludes “hollow” segments just consisting of their boundaries. It thus says that the relations B and B^+ are *dense*. Hence it may be called the *denseness axiom*; cf. [1, p. 146].²² It corresponds to the second of Hilbert's “axioms of order”; cf. [12, chap. I, §3].²³

Within our framework, then, we can prove Pasch's VIII. by means of the principles of decomposability, ramification and denseness. Actually VIII. refers to a special constellation considered in the straightness axiom *GEO 6*. If in that axiom $s_1 = p_3p_1$ and $s_2 = p_2p_4$, then we expect that $s_1 + s_2 = p_3p_4$; cf. the left hand side of Fig. 4.

The 24: $p_1 \neq p_2 \wedge p_2Bp_3p_1 \wedge p_1Bp_2p_4 \rightarrow$
 $p_3p_1 + p_2p_4 = p_3p_4$

Proof: According to *GEO 6*, $p_3p_1 + p_2p_4$ is a segment, and according to *The 8* and *The 15* it contains the segment p_3p_4 as a part. Thus it remains to be shown that also conversely $p_3p_1 + p_2p_4 \in P^>p_3p_4$ (*). — In order to prove this, we first deal with some special cases. (a) If $p_1 = p_3$, then $p_1 = p_3p_1 = p_3$. But this would, in contradiction to the assumption of the theorem, imply that $p_1 = p_2$ since $p_2Bp_3p_1$. Hence $p_1 \neq p_3$. — (b) Furthermore, $p_2 \neq p_4$, too. For otherwise we would have $p_2 = p_2p_4 = p_4$ and hence from the hypothesis $p_1Bp_2p_4$ $p_1 = p_2$, which again contradicts the assumption $p_1 \neq p_2$. — (c) Finally, $p_3 \neq p_4$, too. For otherwise we had $p_1Bp_2p_3$ because of $p_1Bp_2p_4$. But since $p_2Bp_3p_1$, $p_3p_1 = p_3p_2 + p_2p_1$ with

²²In [1], that axiom is formulated for a segment operation which maps pairs of points p_1 and p_2 to the *open* segment bounded by those two points; cf. Fn. 3. This means that the points do not belong to the segment which they delimit.

²³Hilbert, however, conceives of order as a relation restricted to the points of some given line. We shall return to the topic of lines at the end of the present subsection.

$p_2 = p_3p_2 \cdot p_2p_1$. As we have just seen, however, $p_1Bp_2p_3$; this would yield $p_1 = p_2$, once more in contradiction to the assumption. (d) We may assume that $p_4 \notin P^>p_3p_1$ (and so particularly $p_1 \neq p_4$). Otherwise, by decomposition, $p_4Bp_3p_2$ (d.1) or $p_4Bp_2p_1$ (d.2). The latter would imply $p_1 = p_4$ by *The 20*. Since the assumption $p_2Bp_3p_1$ implies $p_2p_4 \in P^>p_3p_1$ by *The 15*, we had $p_3p_1 + p_2p_4 = p_3p_1$ which together with $p_1 = p_4$ implies (*), which concludes the case (d.2). — Assume then (d.1). By decomposition $p_3p_4 + p_4p_2 = p_3p_2$ with $p_4 = p_3p_4 \cdot p_4p_2$. From this and from (b) above we conclude that $p_2 \notin P^>p_3p_4$. Again by decomposition $p_3p_4 + p_4p_1 = p_3p_1$; thus $p_2Bp_4p_1$. But then with the assumption $p_1Bp_2p_4$ and *The 20* $p_1 = p_2$ in contradiction to the theorem's assumption. — (e) Because of a similar reasoning we may also assume that $p_3 \notin P^>p_2p_4$ (and thus in particular $p_2 \neq p_3$). — In order to prove (*), it is sufficient to show that $p_1, p_2 \in P^>p_3p_4$. From the latter and *The 15* it follows that p_3p_1 and p_2p_4 are both parts of p_3p_4 which implies (*) by *The 3*. — Assume first that p_1 were not a part of p_3p_4 . We may take it for granted that $p_4 \notin P^>p_3p_1$; cf. (d) above. Hence we may apply *The 21* and conclude that $p_3 = p_3p_4 \cdot p_3p_1$ (f). Now consider p_2p_4 . From (f), the assumption $p_2Bp_3p_1$, and the fact that — according to (b), (c) above — p_2 differs from both p_3 and p_4 , we infer that $p_2 \notin P^>p_3p_4$. Furthermore we may assume that $p_3 \notin P^>p_2p_4$; cf. (e) above. Thus we may once again by *The 21* conclude that $p_4 = p_3p_4 \cdot p_2p_4$ (g). By (f) and (g), we have that p_3 and p_4 are the only points shared by p_3p_4 and $p_3p_1 + p_2p_4$. We had already proven that $p_3p_4P^>p_3p_1 + p_2p_4$ and hence $p_3p_4(p_3p_1 + p_2p_4) = p_3p_4$. But since the factors of that product share only the two points mentioned, we are forced to conclude that $p_3p_4 = p_3 + p_4$ which contradicts denseness. — The same result is reached by a parallel argumentation which starts from the assumption that $p_2 \notin P^>p_3p_4$.

Pasch's VII. is a consequence of the ramification property; cf. *The 21*.

The 25: $p_2B^+p_1p_3 \wedge p_2B^+p_1p_4 \rightarrow p_3B^+p_1p_4 \vee p_4B^+p_1p_3$

Proof: Since we are concerned with strict betweenness here, $p_1 \neq p_2$. Hence the two segments p_1p_3 and p_1p_4 share more than just one point; thus $p_1 \neq \inf(p_1p_3, p_1p_4)$. By the ramification property then: $\neg[p_4 \notin P^>p_1p_3 \wedge p_3 \notin p_1p_4]$, i.e., $p_3 \in P^>p_1p_4 \vee p_4 \in P^>p_1p_3$, hence (since, by hypothesis, both $p_1 \neq p_3$ and $p_1 \neq p_4$): $p_3B^+p_1p_4 \vee p_4B^+p_1p_3$. ■

We have thus proven (or, in the case of II. and VI. simply taken over) all of Pasch's axioms for lines. The celebrated axiom called after him ("Pasch's Axiom"; Hilbert's fourth "axiom of order"), however, is not included within the list of Tab. I because it makes its appearance in [51] only in the book's second paragraph, which deals with planes. We shall return to Pasch's Axiom in the next section. We conclude the present section by a brief consideration of (unbounded) lines (as opposed to bounded segments).

Pasch [51, p. 4] rejects the notion of an infinitely extended line since it does not "correspond to anything perceivable". Nevertheless, he introduces lines into his system by a special

procedure which he calls "implicit definition" (and is to be distinguished from "definition by axioms" also called thus). The procedure is more closely described in [52], where Pasch relates it to the doctrine of the As-If of the neo-Kantian philosopher Hans Vaihinger. It essentially consists in the introduction of a kind of new, "fictitious" objects which, however, can be uniquely characterized by their relationships to "real" objects. In our mereological framework we construct lines as sums of certain points. The following definition mirrors the set-theoretical procedure of [53, p. 50].

Def 17: $L \stackrel{\text{def}}{=} \{ \langle u_1, p_1, p_2 \rangle \mid p_1 \neq p_2 \wedge u_1 = \sup(p_3 \mid p_1Bp_3p_2 \vee p_3Bp_1p_2 \vee p_2Bp_1p_3) \}$

Obviously, L is a (partial) operation. Coppel [53, p. 49–52] develops a theory of lines based on four axioms (L1)–(L4). (L1) is the idempotent law *The 16*. Coppel's (L2) says that $p_2Bp_1p_4$ if $p_2Bp_1p_3$, $p_3Bp_2p_4$, and $p_2 \neq p_3$. This is an immediate consequence of our *The 24*. (L3) is the ramification principle; cf. *The 21*. Finally (L4) is that part of *GEO 5* which is concerned with the supremum-operation (omitting the part postulating that the combined segments share a single point). Since all of Coppel's axiom are provable in our framework (and his proofs do not make use of set-theoretical procedures not available in our class-theoretical framework), the theorems proven by him can directly be taken over into our framework. This includes his result that a line is uniquely determined by any two points lying on it and that the points on a line can be arranged in a unique way in a total linear order. Because of *GEO 8*, we may add that this order is dense.

C. The Axioms of Peano and Pasch

The structure of points and lines described at the end of the previous section is known as a "line space"; cf. [1, p. 155]. In a line space two distinct points belong to a unique line, each line has at least two points, and the points on each line are arranged in a total linear dense order. The only condition entering into the definition of a line space which has not been established yet is the so-called *Peano Axiom* (cf. the left-hand side of Fig. 6). However, this axiom can be proven in our framework.

The 26: $p_4Bp_2p_3 \wedge p_5Bp_1p_4 \rightarrow \exists p_6 \in P^>p_1p_2 \cdot p_5Bp_3p_6$

Proof: Suppose that both $p_4Bp_2p_3$ and $p_5Bp_1p_4$. Then p_5 belongs to $[p_1, p_2, p_3]$ which, by *GEO 4*, is $\sup(s \mid \exists p_4 \in P^>p_2p_3 \cdot s = p_1p_4)$. By the same axiom, however, $[p_1, p_2, p_3] = \sup(s \mid \exists p_6 \in P^>p_1p_2 \cdot s = p_3p_6)$. Hence $p_5 \in P^>p_3p_6$ for some $p_6 \in P^>p_1p_2$. ■

From the Peano Axiom we now deduce Pasch's Axiom. Pasch himself formulates his axiom [51, p. 20] as stating a relationship between a triangle and a segment entering the inner of the triangle by crossing one of its edges. The theorem then says that either the entering segment itself or a prolongation of it leaves the triangle by crossing (either a vertex of the triangle or) exactly one of the two other edges. Here we prove another formulation. Pasch's own version of

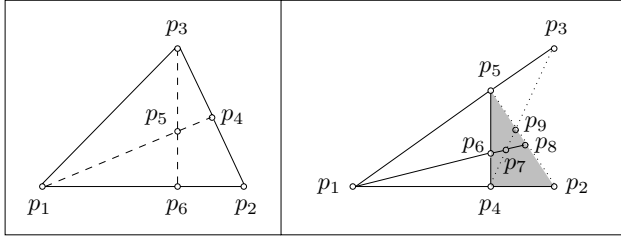


Fig. 6. The Peano Axiom and Proof of Pasch's Axiom

his axiom, however, is provable from the version used here, cf. [53, p. 57f]. Conversely, our version given below is implied by Pasch's theorem; cf. [51, p. 25].

The 27: $p_4Bp_1p_2 \wedge p_5Bp_1p_3 \rightarrow \langle p_2p_5, p_3p_4 \rangle \in O$

Proof: We²⁴ may presuppose that none of the three points p_1, p_2, p_3 lies on the segment determined by the two others. For in that case all the points involved lie on that segment. In this case the theorem is easily proven by checking a number of trivial possibilities. By *GEO 8* there exists a point p_6 between p_5 and p_4 . Obviously, $p_1 \neq p_6$ since otherwise p_1, p_2, p_3 would lie on a common segment contrary to our initial assumption. For p_6 two applications of *The 26* yield points p_7 and p_8 on, respectively, p_3p_4 and p_2p_5 such that both $p_6Bp_1p_7$ and $p_6Bp_1p_8$. By *GEO 6* p_1, p_6, p_7 , and p_8 lie on a segment; hence, by *The 22* one of the three points p_6, p_7 , and p_8 must lie within the segment spanned by the two others. If $p_6Bp_7p_8$, it would follow that all three points were identical and then that point would be a common point of p_2p_5 and p_3p_4 . Hence $p_7Bp_6p_8$ or $p_8Bp_6p_7$. The first case is that displayed in the left hand side of Fig. 6. We consider only this case; the argument for the other case is completely analogous. The segment p_6p_8 is a part of the triangle $\triangle p_4p_5p_2$ (shaded in Fig. 6); therefore the point p_7 also belongs to that triangle. It follows by an application of *GEO 4* that $p_7Bp_4p_9$ for some p_9 with $p_9Bp_2p_5$. Applying now *The 26* to $\triangle p_1p_2p_3$ and the point p_9 lying on the segment p_2p_5 we infer that there must be a point p_{10} between p_1 and p_2 such that $p_9Bp_3p_{10}$. Furthermore, $p_9 \neq p_3$, for else p_1, p_2, p_3 would again lie on a single segment which is impossible according to our initial assumption. The segments p_3p_{10} and p_3p_4 share thus two points, namely p_3 and p_9 . Hence, by *The 6*, the sum s is a segment. This segment s contains both p_4 and p_{10} . If these were two different points, then, again by *The 6*, the sum of s with p_1p_2 would be a segment containing all three of p_1, p_2, p_3 . Hence we conclude that $p_4 = p_{10}$. But then $p_9Bp_3p_4$ and p_3p_4 and p_2p_5 overlap in p_9 . ■

We conclude the presentation of our theory of space by some remarks relating to two topics addressed in the INTRODUCTION (sec. I), namely “coordinates” and the idea of a “geometric algebra”.

²⁴Van de Vel [1, p. 144] gives another proof of Pasch's Axiom making use only of the Ramification Principle instead of the stronger Straightness Principle. The proof given here employs the same idea as is used by Coppel [53, p. 86] but applies *GEO 4* in order to infer the existence of point p_9 .

V. COORDINATES

The first step in turning a qualitative theory of space into a “quantitative” one by the introduction of coordinates is to define operations of addition and multiplication for the segments of a line (or, when a certain point of the line is distinguished as the “origin” or “zero point”, for the points of that line) by help of Desargue's theorem; cf., e.g., [12, § 24] or [50, Part I, § 14]. Now Coppel [53, p. 54f] defines a “linear geometry” as a structure fulfilling (set-theoretic counterparts) of Pasch's principle VIII. (which follows from our *The 24*),²⁵ our *The 21* (the “ramification principle”), our *GEO 5* (the “decomposability principle”) and the Peano Axiom (our *The 26*). Furthermore, he shows [53, pp. 125–131] that in each dense linear geometry (thus in each linear geometry in which our *GEO 8* is valid) of dimensionality greater than 2, Desargue's theorem holds true in the following form: Let $L(p_1p_2)$, $L(p_3p_4)$, $L(p_5p_6)$ three distinct lines with a common point p which is different from p_j ($1 \leq j \leq 6$). If corresponding sides of the two triangles $\triangle p_1p_3p_4$ and $\triangle p_2p_4p_6$ intersect, then the three points of intersection lie on the same line. Finally, it is shown by Coppel [53, ch. 7] that suitable additive and multiplicative operations for points can be defined by use of Desargue's theorem in such a way that a linear geometry in which this theorem is valid may be embedded into (the projective completion of) an ordered skew field. Thus in order to introduce coordinates we can either postulate that our space is three-dimensional or we may directly require that Desargue's theorem is true.

Of course, if it is desired that the order of points on a line have Dedekind's cut property something more is necessary. In the common axiomatisation of geometry the axiom of continuity guaranteeing the cut property makes use of quantification over sets of points which is not available in LC. However, since regions are just the sums of their points, first-order quantification over regions mimics quantification over point sets. Thus it would be interesting to see how far one gets with something like $\exists p. \forall p_1 \in P^{>u_1}, p_2 \in P^{>u_2}. p_1Bpp_2 \rightarrow \exists p_0 \forall p_1 \in P^{>u_1}, p_2 \in P^{>u_2}. pBp_1p_2$ which results from Tarski's axiom on continuity [50, p. 14] by replacing variables for point sets by variables for regions.

VI. NOTE ON THE IDEA OF A “GEOMETRIC ALGEBRA”

In the INTRODUCTION (sec. I) we considered Leibniz' idea of a geometric algebra in which one can directly calculate with points and lines without encoding these geometric entities by (pairs and sets of) numbers. Furthermore, the geometric system presented here has been inspired in many respects by Prenowitz' “join geometry” which shares the “algebraic vision” with Leibniz' idea of a *characteristica geometrica*. However, that system with its axioms involving logical connectives and quantifiers is more in line with the ancient Euclidean procedure than with modern algebraic theories—

²⁵Actually, not VIII. itself but an equivalent “mirror image” of it in which the sequence of points concerned is inverted is used by Coppel.

like, e.g., ring theory or lattice theory — which fix their models by lists of equations.

Nevertheless there is the possibility to develop within our framework parts of geometry in an “equational manner”. In the present section we briefly describe an example for this. In the INTRODUCTION (sec. I) it was pointed out that the “associative law” $p_1(p_2p_3) = (p_1p_2)p_3$ for Prenowitz’ join operation is only interpretable on the basis of certain conventions. The problem is that the join operation is defined for points while, one subterm (namely, respectively “ (p_2p_3) ” and “ (p_1p_3) ”) of the main terms on both side of the equation stating the law refers to a segment rather than to a point. In the framework presented in this article this is not really a problem. We have just to replace Def 14-(c) by the two definitions $p_1u_1 \stackrel{\text{def}}{=} u_1p_1 \stackrel{\text{def}}{=} [p_1, u_1]$. Since the universal variable “ u_1 ” is substitutable by variables of both sort **p** and sort **s**, this simultaneously defines the operations of (1) joining points, (2) joining a point to a segment, and (3) joining a segment to a point; and their would be no problem with a terms like “ $p_1(p_2p_3)$ ” and “ $(p_1p_2)p_3$ ” entering into the formulation of the associative law.

However, this solution suffers from two shortcomings. (1st) It seems too restrictive by requiring that one argument of the generalized join operation is still a point. Should one not define the operation in a completely general way admitting for arguments of any sort? (2nd) It does not match Prenowitz’ intention that the join of a point and a segment is built up by the segments joining the point argument and the individual points of the segment argument. Thus it would be more adequate to keep the old Def 14-(c) for joining points and to supplement it by the following general one.²⁶

Def 18: $u_1 \circ u_2 \stackrel{\text{def}}{=} \sup(s \mid \exists p_1 \in P^>u_1, p_2 \in P^>u_2. s = p_1p_2)$

Obviously, $p_1 \circ p_2 = p_1p_2$; we do not therefore make a notational distinction between the two operations. It is immediate that the general join operation is commutative. Furthermore, it is obvious that it is idempotent for convex regions. Thus two of the algebraic laws valid for his special join operation hold true for the more general one when it is restricted to convex regions.

The 28: Idempotency $cc = c$
Commutativity $c_1c_2 = c_2c_1$

Is the third law postulated to hold by Prenowitz for his join operation, thus associativity, valid (for convex regions), too? In order to show that $c_1(c_2c_3) = (c_1c_2)c_3$, it suffices in view of The 6 to show that $c_1(c_2c_3)$ and $(c_1c_2)c_3$ have exactly the same points. Let us consider a point p_4 of $c_1(c_2c_3)$; we have to show that $p_4 \in P^>(c_1c_2)c_3$. From $p_4P^>c_1(c_2c_3)$ we conclude the existence of points $p_j \in P^>c_j$ ($1 \leq j \leq 3$) and of a point $p_5Bp_2p_3$ such that $p_4Bp_1p_5$; cf. the constellation built up by the solids line in the left hand diagram of Fig. 7. That p_4

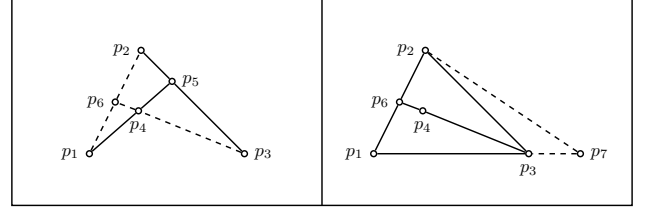


Fig. 7. Associativity of the generalized join operation

belongs to $(c_1c_2)c_3$ means that there are points $p'_j \in P^>c_j$ $1 \leq j \leq 3$ and a point p'_5 such that $p'_5Bp'_1p'_2$ and $p_4Bp'_5p'_3$. Considering the situation depicted on the left hand side of Fig. 7, it suggests itself to set $p'_j := p_j$ $1 \leq j \leq 3$ and to chose p'_5 as the point of intersection p_6 of p_1p_2 and the line through the segment p_3p_4 ; cf. the constellation of dashed lines in the left part of Fig. 7.

What then remains to be shown is that p_6 really exists, i.e., that the segment p_3p_4 can be extended so that it intersects with p_1p_2 . But one readily recognizes that the situation described is just a “Pasch-configuration”. Using GEO 7, we may extend p_1p_3 to a point p_7 such that $p_3Bp_1p_2$. Then the line through p_3p_4 is a line entering the triangle $\Delta p_1p_7p_2$ through its edge p_1p_7 . The original version of Pasch’s Axiom requires that this line leaves the triangle through one of the two other edges. In our case p_6 must be incident with p_1p_2 . Assume that p_6 were a part of p_7p_2 . The only point that p_7p_2 has in common with $\Delta p_1p_3p_2$ is p_2 ;²⁷ hence we had $p_2 = p_6$ and thus nevertheless $p_6 \in p_1p_2$. So we have proven that $p_4 \in P^>(p_1p_2)p_3$ and by this that $P^>p_1(p_2p_3) \subseteq P^>(p_1p_2)p_3$. The converse of this can be shown by an analogous argument. Summing up, we have thus proven associativity.

The 29: Associativity $c_1(c_2c_3) = (c_1c_2)c_3$

Prenowitz’ [9, p. 55] three basic algebraic laws (J2), (J3), and (J4) of his join geometry are (*modulo* the difference explained in Fn. 3 above) just special cases of the more general principles given here and can be derived from these principles because of MER 2.

VII. CONCLUSION

We have provided a theory of space formulated in a mereological framework which is based on the notion of convexity. Using mereological concepts, segments of straight lines have been explained as the convex hulls of the sum of two points. What “straightness” exactly means for thus defined segments has been determined by two axioms. Two further axioms have been included into the system in order to describe the linear arrangement of the points of a segment. Finally, it has been illustrated by examples how a more algebraic approach to geometry, envisaged already by Leibniz, can be developed within the framework presented here.

²⁶Def 18 corresponds in our mereological framework to a similar definition provided by Prenowitz and Jantocziak in their set-theoretic setting; cf. [9, p. 55].

²⁷In order to see this, one has to use GEO 6.

ACKNOWLEDGMENT

The authors would like to thank Carola Eschenbach, Christopher Habel, Felix Lindner, and Johanna Seibt for a series of very fruitful discussions in the Danish-German research initiative *Making Space*.

REFERENCES

- [1] M. L. J. van der Vel, *Theory of Convex Structures*. Amsterdam: North-Holland Publishing Company, 1993.
- [2] Euclid, *The Thirteen Books of the Elements*. 3 vols., 2nd ed. New York: Dover, 2000.
- [3] H. Zeuthen, *Geschichte der Mathematik im Altertum und Mittelalter*. Copenhagen: Høst, 1896.
- [4] G. W. Leibniz, *Mathematische Schriften*. Ed. C. J. Gerhardt. Hildesheim and New York: Olms, 1962, vol. 5, ch. De analysi situ, pp. 178–183, reprint of the original edition from 1858.
- [5] V. Debuiche, “Perspective in Leibniz’s invention of Characteristica geometrica: the problem of Desargues’ influence,” *Historia Mathematica*, vol. 40, pp. 359–385, 2013. doi: 10.1016/j.hm.2013.08.001
- [6] S. C. J. and P. Schreiber, *5000 Jahre Geometrie. Geschichte, Kulturen, Menschen*. Berlin etc.: Springer, 2001.
- [7] H. Grassmann, *Gesammelte mathematische und physikalische Werke. Vol. I. Part I*. Leipzig: Teubner, 1894, reprinted New York: Chelsea Pub. Co., 1969.
- [8] W. Prenowitz, “Projective geometries as multigroups,” *American Journal of Mathematics*, vol. 65, pp. 235–256, 1943.
- [9] W. Prenowitz and J. Jantosciak, *Join Geometries. A Theory of Convex Sets and Linear Geometry*. Berlin etc.: Springer, 1979.
- [10] W. V. O. Quine, “New foundations for mathematical logic,” *American Mathematical Monthly*, vol. 44, pp. 70–80, 1937, reprinted as Chapter V in W. V. O. Quine: *From a Logical Point of View*. Cambridge MA: Harvard University Press, 1961.
- [11] A. Schmidt, “Über deduktive Theorien mit mehreren Sorten von Grunddingen,” *Mathematische Annalen*, vol. 115, pp. 485–506, 1938.
- [12] D. Hilbert, *Grundlagen der Geometrie*. Leipzig: Teubner, 1899, 14th edition. Edited and supplied with appendices by Michael Toepell. Stuttgart etc.: Teubner. — English translation: *Foundations of Geometry*. LaSalle IL: Open Court 1971. 10th printing 1999.
- [13] J.-M. Glubrecht, A. Oberschelp, and G. Todt, *Klassenlogik*. Mannheim: BI Wissenschaftsverlag, 1983.
- [14] P. M. Gruber and W. J. M., *Handbook of Convex Geometry*. 2 vols. Amsterdam: North-Holland Pub. Co., 1993.
- [15] J. O’Rourke, *Computational Geometry*, 2nd ed. Cambridge GB: Cambridge University Press, 1998.
- [16] P. Gärdenfors, *Conceptual Spaces. The Geometry of Thought*. Cambridge MA: MIT Press, 2000.
- [17] D. A. Randell, Z. Cui, and A. G. Cohn, “A spatial logic based on regions and connections,” in *Principles of Knowledge Representation and Reasoning. Proceedings of the 3rd International Conference*, B. Nebel, C. Rich, and W. Swartout, Eds. Los Altos CA: Morgan Kaufmann, 1992, pp. 165–176.
- [18] I. Pratt, “First-order qualitative spatial representation languages with convexity,” *Spatial Cognition and Computation*, vol. 1, pp. 181–204, 1999.
- [19] A. Trybus, “An axiom system for a spatial logic with convexity,” Ph.D. dissertation, University of Manchester, Manchester, 2011.
- [20] D. Lewis, *Parts of Classes*. Oxford: Blackwell, 1991.
- [21] R. Urbaniak, *Leśniewski’s Systems of Logic and Foundations of Mathematics*. Cham: Springer, 2014.
- [22] R. E. Clay, “Relation of Leśniewski’s mereology to Boolean algebra,” *The Journal of Symbolic Logic*, vol. 39, pp. 638–648, 1974.
- [23] A. Tarski, “Les fondements de la géométrie des corps,” *Księga Pamiątkowa Pierwszego Polskiego Zjazdu Matematycznego (Supplement to Annales de la Société Polonaise de Mathématique)*, pp. 29–33, 1929, English translation in: Tarski, Alfred (1983): *Logic, Semantics, Metamathematics*. 2nd edition ed. by John Corcoran. Indianapolis IN: Hackett. 24–29.
- [24] A. Grzegorzczak, “Axiomatizability of geometry without points,” *Synthese*, vol. 12, pp. 228–235, 1960.
- [25] B. L. Clarke, “Individuals and points,” *Notre Dame Journal of Formal Logic*, vol. 26, no. 61–75, 1985. doi: 10.1305/ndjfl/1093870761
- [26] R. Gruszczyński and A. Pietruszczak, “Full development of Tarski’s geometry of solids,” *The Bulletin of Symbolic Logic*, vol. 14, pp. 481–540, 2008. doi: 10.2178/bsl/1231081462
- [27] T. Hahmann, M. Winter, and M. Gruninger, “Stonian p-ortholattices: A new approach to the mereotopology RT_0 ,” *Artificial Intelligence*, vol. 173, pp. 1424–1440, 2009. doi: 10.1016/j.artint.2009.07.001
- [28] R. Casati and A. C. Varzi, *Parts and Places*. Cambridge MA: MIT Press, 1999.
- [29] A. N. Whitehead, *An Enquiry Concerning the Principles of Natural Knowledge*. Cambridge GB: Cambridge University Press, 1919.
- [30] —, *The Concept of Nature. The Turner Lectures Delivered in Trinity College 1919*. Cambridge GB: Cambridge University Press, 1920.
- [31] —, *Process and Reality. An Essay in Modern Cosmology*. New York: McMillan, 1929, paperback edition 1969.
- [32] Aristotle, *The Metaphysics*, 2nd ed. London: Penguin, 2004.
- [33] Proclus, *A Commentary on the First Book of Euclid’s Elements*. Princeton NJ: Princeton University Press, 1992.
- [34] R. Stroh, *Die Grundbegriffe der reinen Geometrie in ihrem Verhältnis zur Anschauung*. Leipzig: Teubner, 1925.
- [35] T. d. Laguna, “Point, line, and surface, as sets of solids,” *Journal of Philosophy*, vol. 19, pp. 449–461, 1922. doi: 10.2307/2939504
- [36] S. Leśniewski, *S. Leśniewski’s Lecture Notes in Logic*. Dordrecht: Kluwer, 1988, ch. 6. Whitehead’s theory of events, pp. 171–178.
- [37] E. A. Marchisotto and J. T. Smith, *The Legacy of Mario Pieri in Geometry and Arithmetic*. Boston etc.: Birkhäuser, 2007.
- [38] J. Renz, *Qualitative Spatial Reasoning with Topological Information*. Berlin etc.: Springer, 1998.
- [39] C. Eschenbach, “A mereotopological definition of ‘point’,” in *Topological Foundations of Cognitive Science. Papers from the Workshop at the FISI-CS, Buffalo, NY, July 9–10, 1994*. Hamburg: Graduiertenkolleg Kognitionswissenschaft, 1994, pp. 63–80.
- [40] G. Gerla, “Pointless geometries,” in *Handbook of Incidence Geometry*, F. Buekenhout, Ed. Amsterdam: Elsevier Science, 1995, pp. 1015–1031.
- [41] K. Menger, “Topology without points,” *Rice Institute Pamphlets* 27, pp. 80–107, 1940.
- [42] G. Dimov and D. Vakarelov, “Contact algebras and region-based theory of space i, ii,” *Fundamenta Informaticae*, vol. 74, pp. 209–249, 251–282, 2006.
- [43] D. Vakarelov, “Region-based space: algebra of regions, representation theory, and logics,” in *Mathematical Problems from Applied Logic II. Logics for the XXIst Century*, D. Gabbay, Goncharov, S. S., and M. Zakharyashev, Eds. New York: Springer, 2007, pp. 267–347.
- [44] T. Hahmann, M. Winter, and M. Gruninger, “On the algebra of regular sets: Properties of representable Stonian p-ortholattices,” *Annals of Mathematics and Artificial Intelligence*, vol. 65, pp. 35–60, 2012. doi: 10.1007/s10472-012-9301-2
- [45] C. Coppola and G. Gerla, “Multi-valued logic for a point-free foundation of geometry,” in *Mereology and the Sciences. Parts and Wholes in the Contemporary Scientific Context*, C. Calosi and P. Graziani, Eds. Springer, 2012, pp. 105–122.
- [46] P. Suppes, *Representation and Invariance of Scientific Structures*. Stanford CA: Center for the Study of Language and Information, 2002.
- [47] E. Rubin, *Synsøplevede Figurer. Studier i psykologisk Analyse. Første Del*. Copenhagen: Gyldendal, 1915.
- [48] O. Selz, “Die Struktur der Steigerungsreihen und die Theorie von Raum, Zeit und Gestalt,” in *Bericht über den XI. Kongress für experimentelle Psychologie in Wien vom 9.–13. April 1929*, H. Volkelt, Ed. Jena: Fischer, 1930, pp. 27–45.
- [49] “Appendix E. An alternative system for **P** and **T**,” In: J. H. Woodger: *The Axiomatic Method in Biology*. Cambridge GB: Cambridge University Press, 1937. 161–172.
- [50] W. Schwabhäuser, Wolfram; Szmielew and A. Tarski, *Metamathematische Methoden in der Geometrie*. Berlin etc.: Springer, 1983.
- [51] M. Pasch, *Vorlesungen über neuere Geometrie*. Leipzig: Teubner, 1882, 2nd edition with an appendix by Max Dehn. Berlin: Springer 1926. Reprinted 1976.
- [52] —, “Die Begründung der Mathematik und die implizite Definition. Ein Zusammenhang mit der Lehre vom Als-Ob,” *Annalen der Philosophie*, vol. 2, pp. 145–162, 1921.
- [53] W. A. Coppel, *Foundations of Convex Geometry*. Cambridge GB: Cambridge University Press, 1998.

Encoding Relative Orientation and Mereotopology Relations with Geometric Constraints in CLP(QS)

Carl Schultz

Institute for Geoinformatics
University of Münster, Germany
Email: schultzc@uni-muenster.de

Mehul Bhatt

Department of Computer Science
University of Bremen, Germany
Email: bhatt@informatik.uni-bremen.de

Abstract—One approach for encoding the semantics of spatial relations within a declarative programming framework is by systems of polynomial constraints. However, solving such constraints is computationally intractable in general (i.e. the theory of real-closed fields), and thus far the proposed symbolic algebraic methods do not scale to real world problems. In this paper we address this intractability by investigating the use of constructive geometric constraint solving (in combination with numerical optimisation) within the context of constraint logic programming over qualitative spaces, CLP(QS). We present novel geometric encodings for relative orientation and mereotopology relations and show that our encodings are significantly more robust than other proposed approaches for directly encoding inequalities, due to our encodings being based on a standard, well known set of relations encoded as quadratic equations. Our encodings are general (i.e. not implementation specific) and can thus be directly employed in *all* standard geometric constraint solvers, particularly solvers that are prominent within the Computer Aided Design and Manufacturing communities. We empirically evaluate our approach on a range of benchmark problems from the Qualitative Spatial Reasoning community, and show that our method outperforms other symbolic algebraic approaches to spatial reasoning by orders of magnitude on those benchmark problems (such as Cylindrical Algebraic Decomposition).

I. INTRODUCTION

Many complex, real world problems can be elegantly expressed in a declarative manner within the paradigm of logic programming. In particular, the user specifies *what* needs to be accomplished, rather than procedurally prescribing *how* the problem should be solved. Often such real world problems inherently involve spatial aspects: variables ranging over spatial object domains (polygons, circles, points, etc.) and spatial relations between those objects.

Qualitative spatial representation and reasoning has received considerable attention, especially from the viewpoint of the research communities of spatial information theory, and knowledge representation and reasoning. Knowledge representation and reasoning about *space* may be formally interpreted within diverse frameworks such as: (a) geometric reasoning & constructive (solid) geometry [20]; (b) relational algebraic semantics of ‘qualitative spatial calculi’ [25]; and (c) by axiomatically constructed formal systems of mereotopology and mereogeometry [1]. Independent of formal semantics, commonsense spatio-linguistic abstractions offer a human-centred and cognitively adequate mechanism for logic-based automated reasoning about spatio-temporal information [4].

However, what is clearly still lacking is a systematic knowledge representation (KR) centred methodological foundation that provides a computational backbone —*formal semantics in the context of mainstream KR methods, declarative (spatial) programming paradigm, general toolsets* – for commonsense reasoning about space. This is essential in order to provide a developmental basis and seamless, systematic integration within large-scale AI, and hybrid intelligent systems that involve diverse types of knowledge representation, reasoning, and learning modules (e.g., IBM Watson).

Our long-term research agenda is to integrate spatial reasoning natively within declarative, logic-based frameworks [3]. We have partially realised aspects of this in the Constraint Logic Programming over Qualitative Spatial domains system: CLP(QS) [3]. Users can specify logic programs that are solved with a seamless combination of spatial reasoning and Prolog’s standard variable unification, i.e. logic programming extended to natively handle qualitative and geometric spatial constraints. Specifically, we utilise the following features of logic programming:

- *declarativeness* - a Prolog program is a specification of the conditions that need to be satisfied, or the goals that must be accomplished, rather than a procedural set of instructions; that is, the user specifies *what* the program should accomplish, rather than *how* the problem at hand should be solved
- *recursive variable generation* - languages such as Prolog are able to infer the existence of objects that are required to solve the problem that were not explicitly specified by the user; combined with recursion, this enables a user to explore the unboundedness and density of objects in space

Relation algebraic qualitative spatial reasoning (e.g. the left-right calculus LR [26]), while efficient, is incomplete in general [21, 22, 25].¹ Alternatively, constraint logic programming based systems such as CLP(QS) [3] and others (see

¹*Incompleteness* refers to the inability of a spatial reasoning method to determine, in general, whether a given set of qualitative spatial constraints is consistent or inconsistent. Relation-algebraic spatial reasoning (i.e. using algebraic closure based on weak composition) has been shown to be incomplete for a number of spatial languages and cannot guarantee *consistency* in general, e.g. relative directions [22] and containment relations between linearly ordered intervals [21], Theorem 5.9.

[6, 7, 28, 29]) adopt an analytic geometry approach where spatial relations are encoded as systems of polynomial equation and inequality constraints. Thus, the task of determining whether a set of spatial constraints is consistent becomes equivalent to determining the satisfiability of a system of polynomial constraints with variables ranging over reals.²

We have investigated a range of polynomial constraint solving methods including CLP(R) for linear constraints, and SAT Modulo Theories and quantifier elimination by Cylindrical Algebraic Decomposition (CAD) for general systems of non-linear constraints [3, 31, 32]. However, solving such constraints is computationally intractable in general (i.e. the theory of real-closed fields; for example, the CAD algorithm has double exponential complexity, $O(a^{b^n})$, in the number of variables in the polynomial constraints, n) [2], and thus far, prominent symbolic algebraic methods do not scale to real world problems [24]. In this paper we address this intractability by investigating the use of constructive geometric constraint solving for spatial reasoning within the context of CLP(QS).

The paper is structured as follows. In the remainder of introduction we highlight some key benefits of the utilisation of geometric constraint solving within constraint logic programming over qualitative spaces. Section II provides an introduction to spatial reasoning by polynomial encodings, and formally specifies the standard geometric constraint language. In Section III we empirically evaluate our approach on a range of well known benchmark problems from the Qualitative Spatial Reasoning community, and show that our system outperforms other symbolic algebraic approaches to spatial reasoning by orders of magnitude. Section IV evaluates our encodings using a range of spatial benchmark problems. We then present our conclusions about the main contributions of the paper and directions for future research.

A. Motivations for utilising geometric constraint solving

Geometric constraint solving for higher-level qualitative spatial reasoning has a range of benefits: problems can have a combination of numerical and qualitative information, the methods scale to real world sized problems (in the order of hundreds of objects), and they can produce consistent and “best found” instantiations for both solved and unsolved problems, respectively. Moreover, at each iteration, the solver produces increasingly better configurations as it attempts to converge on the solution; we can directly study and visualise these intermediate configurations as a spatial *history* of configurations [17, 18] giving further insight into the nature of the problem at hand. However, a key limitation is that

²Tarski famously proved that the theory of real-closed fields is decidable via quantifier elimination (see [2, 12, 13] for an overview and algorithms); i.e. in a finite amount of time we can determine the consistency (or inconsistency) of any formula consisting of quantifiers (\forall, \exists) over the reals, and polynomial equations and inequalities combined using logical connectors (\wedge, \vee, \neg). Thus, by encoding spatial relations as systems of polynomial constraints (i.e. analytic geometry) we can employ polynomial constraint solving methods that are guaranteed to determine (in)consistency, giving us sound and complete spatial reasoning.

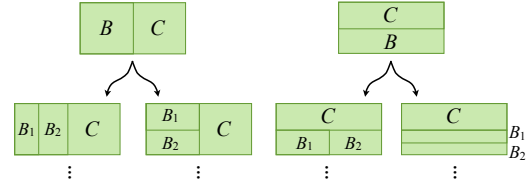


Fig. 1. Recursively enumerating the ways that a rectangle A can be partitioned into a union of rectangles, $\text{union}(B, C)$.

these methods do not handle inequalities well [16], which are used for encoding a large range of qualitative spatial relations, including relative orientation and mereotopology.

Building on results within the geometric constraint community, we present novel geometric encodings for relative orientation and topology relations using only equations (i.e. corresponding to robust ruler and compass construction methods [27]), and thus avoid problematic *saddles* near local optima that are common with inequality encodings using numerical optimisation methods [16]. Moreover, as our encodings rely on a standard set of geometric constraints, they can be used to enhance *all* standard geometric constraint solvers with a variety of qualitative spatial relations that are prominent within the Computer Aided Design and Manufacturing communities.

B. An example of Spatial Reasoning in CLP(QS)

Using recursive variable generation we can declaratively explore the ways that a rectangle can be partitioned into a set of rectangles (employing CLP(QS) mereology relations between boolean combinations of rectangles) [32]:

```
partition(A, union(B,C)) :-
    mereology(equal, A, union(Br,Cr)),
    mereology(discrete_from, Br, Cr),
    ((B = Br, C = Cr)
    ;
    partition(Br, B),
    partition(Cr, C)
    ).

?- partition(rectangle(,_,_), union(B,C)).
```

Importantly, the initial specification of the partitioning task is completely *qualitative* and does not contain any numerical information. We can continually request further solutions from Prolog to generate new solutions. Each solution is defined by a set of qualitative constraints between rectangles and thus represents an infinite set of rectangle configurations. For example, in Figure 1 the left solution in the first row is defined by the constraints: $A_w = B_w + C_w$, $A_h = B_h = C_h$, $A_x = B_x$, $B_x < C_x$ where A_w, A_h is the width and height of A , and (A_x, A_y) is the coordinate of the bottom left corner of A . For each qualitatively distinct solution we can then generate a concrete, numerically defined instantiation of the objects using geometric constraint solving.

We can further constrain the problem to enumerate the ways that a *square* can be partitioned into n squares, $n > 1$. As illustrated in Figure 2, CLP(QS) determines that solutions exist for $n = 4, 6, 7, 8, 9$ and no solutions exist for $n = 2, 3, 5$.

Fig. 2. Square partitioning solutions for $n = 4, 6, 7, 8, 9$.

```

all_squares(square(_,_)).
all_squares(union(A,B)) :-
    all_squares(A), all_squares(B).

?- A=square(_,_),
    partition(A, union(B,C)),
    all_squares(union(B,C)).

```

II. SPATIAL REASONING BY POLYNOMIAL ENCODINGS

Analytic geometry methods parameterise classes of objects and encode spatial relations as systems of polynomial equations and inequalities [10]. For example, we can define a sphere as having a 3D centroid point (x, y, z) and a radius r , where x, y, z, r are reals. Two spheres s_1, s_2 *externally connect* or *touch* if

$$(x_{s_1} - x_{s_2})^2 + (y_{s_1} - y_{s_2})^2 + (z_{s_1} - z_{s_2})^2 = (r_{s_1} + r_{s_2})^2 \quad (1)$$

If the system of polynomial constraints is satisfiable then the spatial constraints are consistent. Specifically, the system of polynomial (in)equalities over variables X is satisfiable if there exists a real number assignment for each $x \in X$ such that the (in)equalities are *true*. Partial geometric information (i.e. a combination of numerical and qualitative spatial information) is utilised by assigning the given real numerical values to the corresponding object parameters.

A. Constructive Geometric Constraint Solving

A plethora of methods have been developed for geometric constraint solving via solving systems of polynomial constraints, and can be broadly categorised as: numerical optimisation (e.g. [16]), symbolic methods (e.g. [10, 14]), and constructive (synthetic) methods (e.g. [8, 15, 27]). We focus on graph-based constructive methods due to their practical efficiency and popularity in industry (e.g. Autodesk Inventor,³ LEDAS LGS2D,⁴ FreeCAD⁵), although our encodings can be similarly applied to any of the aforementioned geometric constraint solving approaches.

In a seminal paper, Owen [27] identifies a particular set of spatial relations that, on one hand, are useful for a wide range of applications, particularly engineering and computer aided manufacturing, and on the other hand, can be reasoned about efficiently enough to address real-world scale problems. The particular set of relations correspond to distance, incidence, and angle constraints that can be encoded as quadratic equations over 2D points, lines, and circles. Geometrically, these correspond to relations that can be constructed using the familiar idealised ruler and compass from Euclid's *Elements* [19]. We refer to this restricted set of spatial relations as the

standard geometric constraint language. This set of relations is now standard within the geometric constraint solving community, and all prominent, industry-standard constraint solvers that we are aware of adopt precisely this set of relations (although sometimes such systems also support ellipses), particularly within Computer Aided Design and Manufacturing.

Owen [27] also presents an influential graph-based reduction method for constructive geometric constraint solving. Spatial information is represented as a graph, where nodes are variables that range over a spatial domain of geometries, and edges represent spatial constraints. Firstly, the graph is analysed, and then a sequence of construction steps is determined that produces a configuration of objects that satisfies the (declarative) geometric constraints.

We emphasise that, as our encodings are based on this standard geometric constraint language, they can be utilised within *all* prominent geometric constraint solvers that also adopt this language. That is, our encodings are not solver-implementation specific.

B. The Standard Geometric Constraint Language

The spatial domains of objects in the standard geometric constraint language are points \mathbf{P} , lines \mathbf{L} , and circles \mathbf{C} :

- a point $p \in \mathbf{P}$ is a pair of reals, $(x_p, y_p) \in \mathbb{R}^2$
- a line $l_{ab} \in \mathbf{L}$ is a pair of distinct points, $a, b \in \mathbf{P}$, $a \neq b$
- a circle $C_i \in \mathbf{C}$ is a circle with centre point $p_i \in \mathbf{P}$ and radius $r_i \in \mathbb{R}$, $0 < r_i$

We use lower case letters to refer to points. We use $l_{p_1 p_2}$ to refer to lines between points in the subscript. We use upper case C_i with a subscript number (if needed) to refer to circles. For brevity, if two points a, b have been declared, then we can refer to the line l_{ab} without explicitly quantifying l , and doing so implicitly constrains a, b to be distinct, e.g. let φ be a predicate, then:

$$\exists a, b \in \mathbf{P}, \exists l_{ab} \in \mathbf{L} (\varphi(l_{ab})) \equiv \exists a, b \in \mathbf{P} (\varphi(l_{ab})).$$

Table I presents polynomial encodings for the standard set of geometric constraints between points, lines, circles. They correspond to:

- *incidence* between points-lines, and points-circles (collinear, coincident);
- *orientation* between lines (parallel, perpendicular);
- *constant distance and angles* for lines and circles (distance between points, radii of circles, angle between points a, b about a reference point p).

III. ENCODING QUALITATIVE SPATIAL RELATIONS USING GEOMETRIC CONSTRAINT LANGUAGES

In this section we present a range of novel encodings that enable us to reason about qualitative spatial relations over extended *regions* (in particular, relative orientation and mereotopology over regions) using traditional geometric constraint solving methods that are restricted to the standard geometric constraint language.

³www.autodesk.com/products/inventor/overview

⁴ledas.com/products/lgs2d/

⁵www.freecadweb.org/

Relation	Polynomial Encoding
collinear (COLL) (point p , line l_{ab})	$(x_b - x_a)(y_p - y_a) = (x_b - y_a)(x_p - x_a)$
coincident (COIN) (point a , circle C_i)	$(x_{p_i} - x_a)^2 + (y_{p_i} - y_a)^2 = r_i^2$
coincident (COIN) (line l_{ab} , circle C_i)	$\text{COIN}_{a,C_i} \wedge \text{COIN}_{b,C_i}$
perpendicular (PERP) (lines l_{ab}, l_{cd})	$(y_b - y_a)(y_d - y_c) = -(x_b - x_a)(x_d - x_c)$
parallel (PARA) (lines l_{ab}, l_{cd})	$(y_b - y_a)(x_d - x_c) = (y_d - y_c)(x_b - x_a)$
vertical (VERT) (line l_{ab})	$x_a = x_b$
length (LEN) (line l_{ab} , constant k)	$(x_a - x_b)^2 + (y_a - y_b)^2 = k^2$
angle (ANG) (points a, b, p , constant θ)	$\theta = \text{atan2}((y_b - y_p), (x_b - x_p)) - \text{atan2}((y_a - y_p), (x_a - x_p))$

TABLE I
POLYNOMIAL ENCODINGS OF GEOMETRIC CONSTRAINT RELATIONS.

A. Minimum distance

A minimum distance circle MDIST_C is a circle C with a diameter at least ε . This encoding provides a means to implement a minimum bound on the diameter of a circle. The constrained circle can then be used to enforce minimum distances between spatial objects. Many prominent geometric constraint solving systems (including Inventor and FreeCAD) do not support this constraint.

As illustrated in Figure 3, the encoding adds a fixed-length chord to the circle (i.e. a line where the endpoints are coincident with the circle). The length of the chord determines the minimum permitted diameter of the circle. The circle diameter is minimised when the chord intersects the centroid of the circle. The circle diameter has no upper bound, that is, the chord can be positioned an arbitrary distance from the centroid; the circle diameter must then increase in order to maintain the constraint that the chord endpoints are coincident with the circle. Additionally, we impose a vertical constraint on the chord to improve the solving efficiency of the encoding (i.e. the effect of the vertical constraint is to eliminate one x variable from the chord).

$$\text{MDIST}(C) \equiv \exists l_{ab} \in \mathbf{L} \left(\text{VERT}(l_{ab}) \wedge \text{COIN}(l_{ab}, C) \wedge \text{LEN}(l_{ab}, \varepsilon) \right)$$

Various numerical optimisation methods do provide *box constraints*, i.e. constant bounds on variables, e.g. limited BFGS-B [9]. In such systems this encoding is redundant, as we can enforce a box constraint on the radius of the circle.

B. Point-segment coincidence

While the *collinear* constraint between points and lines is common in geometric constraint systems (i.e. a point lies anywhere on an infinite line), the ability to constrain a point to lie coincident on a line *segment* (i.e. between two points) is typically not supported. The following encoding realises a *coincidence* constraint between a point p and a segment l_{ab} .

As illustrated in Figure 4, firstly, the encoding adds a circle C_1 such that the two endpoints of the given line segment l_{ab} are made coincident to C_1 , and the centroid of the circle is

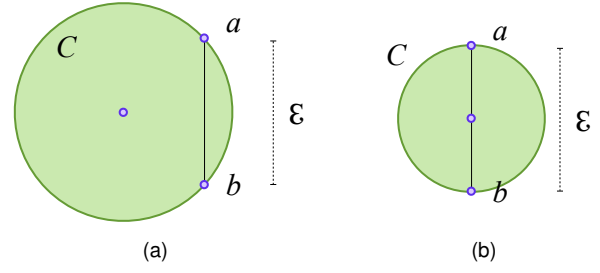


Fig. 3. Minimum distance circle C can not have diameter less than ε .

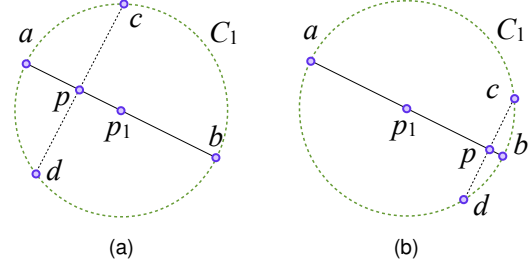


Fig. 4. Point p is constrained to lie on the segment between points a, b .

made collinear with l_{ab} ; in Section III-D we formalise this useful construction between a circle and line, and refer to it as a *brace* relation. Importantly, due to the brace relation, the length of l_{ab} is equal to the diameter of the circle. Next, the encoding adds a line l_{cd} with endpoints coincident to C_1 and perpendicular to l_{ab} . The perpendicular constraint ensures that the two lines always intersect within the interior of C_1 . Finally, as the non-parallel lines necessarily intersect at a single point, the given point p is constrained to be collinear to both lines, and is thus always constrained to lie on the segment l_{ab} .

$$\begin{aligned} \text{COIN}(p, l_{ab}) \equiv & \exists C_1 \in \mathbf{C} \left(\text{COIN}(l_{ab}, C_1) \wedge \text{COLL}(p, l_{ab}) \wedge \right. \\ & \left. \exists l_{cd} \in \mathbf{L} \left(\text{COIN}(l_{cd}, C_1) \wedge \text{PERP}(l_{ab}, l_{cd}) \wedge \right. \right. \\ & \left. \left. \text{COLL}(p, l_{ab}) \wedge \text{COLL}(p, l_{cd}) \right) \right) \end{aligned}$$

For convenience and brevity we also define the relation that segment l_{ab} is coincident with segment l_{cd} as: the endpoints a, b are coincident with the segment l_{cd} .

$$\text{COIN}(l_{ab}, l_{cd}) \equiv \text{COIN}(a, l_{cd}) \wedge \text{COIN}(b, l_{cd})$$

Points p can not be equal to either endpoint a, b as the line l_{cd} can not have zero length. If we drop this line constraint for l_{cd} so that c, d can also be equal, then p can also equal the endpoints. These are useful predicates for defining topological relations in Section III-D, and thus we refer to them as: $\text{COIN}^\subseteq(p, l_{ab})$ and $\text{COIN}^\subseteq(l_{ab}, l_{cd})$.

C. Relative Orientation

To the best of our knowledge, no geometric constraint solver is able to directly express qualitative orientation (*left, right*) between line segments and regions (nor points). This stems from the inability of standard solvers to robustly express inequalities. That is, one common polynomial encoding of

relative orientation is using the sign of the cross product to determine the orientation of a point with respect to a directed line segment, i.e. a point p is left of the segment a, b according to the following inequality:

$$p \text{ left of } l_{ab} \equiv_{\text{def}} (x_b - x_a)(y_p - y_a) > (y_b - y_a)(x_p - x_a)$$

As reported by [16], a “trick” in numerical optimisation of introducing a new variable to express the inequality fails, as it produces a *saddle* near the optimum, thus making the problem significantly more difficult to solve. We have confirmed this result using the state-of-the-art numerical optimisation method: limited LBFGS-B [9].⁶

We have identified the following geometric encoding of relative orientation using the standard geometric constraint language; as such, it is supported, and robustly solved for, by all prominent geometric constraint solvers (Inventor, FreeCAD, LEDAS).

As illustrated in Figure 5, the encoding for the *left of* relation adds a new point c collinear to the given line l_{ab} , and adds a line l_{cp} , between the given point p and the new point c . The encoding then adds the constraint that the angle between l_{ab} and l_{cp} is 90° counter-clockwise. The length of the line l_{cp} is unbounded, and thus p can be moved an arbitrary distance away from l_{ab} . The key is that, if p is moved to the right side of l_{ab} , then the angle constraint is violated, and thus p is forced to remain on the left side.

$$\begin{aligned} \text{LEFT}(p, l_{ab}) &\equiv \\ &\exists c \in \mathbf{P} \left(\text{ANG}(b, p, c, \frac{\pi}{2}) \wedge \text{COLL}(c, l_{ab}) \right) \\ \text{RIGHT}(p, l_{ab}) &\equiv \\ &\exists c \in \mathbf{P} \left(\text{ANG}(b, p, c, -\frac{\pi}{2}) \wedge \text{COLL}(c, l_{ab}) \right) \end{aligned}$$

We extend this definition to relative orientation relations between lines and circles (Figure 6(a)).

$$\begin{aligned} \text{LEFT}(C_1, l_{ab}) &\equiv \\ &\exists c, d \in \mathbf{P} \left(\text{ANG}(b, p_1, c, \frac{\pi}{2}) \wedge \text{COLL}(c, l_{ab}) \wedge \right. \\ &\quad \left. \text{COIN}(d, C_1) \wedge \text{COIN}(d, l_{cp_1}) \right) \\ \text{RIGHT}(C_1, l_{ab}) &\equiv \\ &\exists c, d \in \mathbf{P} \left(\text{ANG}(b, p_1, c, -\frac{\pi}{2}) \wedge \text{COLL}(c, l_{ab}) \wedge \right. \\ &\quad \left. \text{COIN}(d, C_1) \wedge \text{COIN}(d, l_{cp_1}) \right) \end{aligned}$$

As illustrated in Figure 6(b), alternative encodings for relative orientation exist that introduce fewer additional constraints and objects per orientation relation, and may also be solved in a more stable way depending on the implementation for the ANG geometric constraint. In the example illustrated in Figure 6(b), a circle C_1 is introduced and constrained to be *left of* the line a, b , and for each point c, d, e , a line is created parallel to l_{ab} with each endpoint c', d', e' coincident to C_1 ; this encoding aims to minimise the use of the ANG constraint when numerous points are constrained to be left of the same line. Identifying the best encodings for particular tasks is a topic of future work.

⁶Implementation available at:
users.iems.northwestern.edu/~nosedal/lbfgsb.html

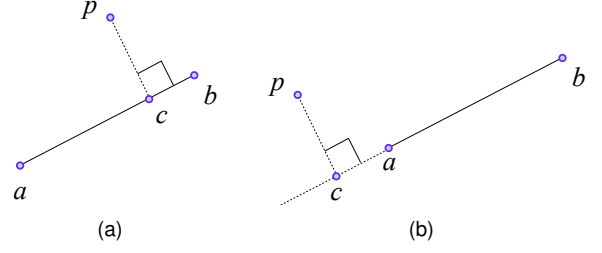


Fig. 5. Point p is constrained to lie anywhere to the *left of* line (a, b) . The angle from point b to p about c is fixed at $\frac{\pi}{2}$ counter-clockwise. The distance between c, p is not constrained.

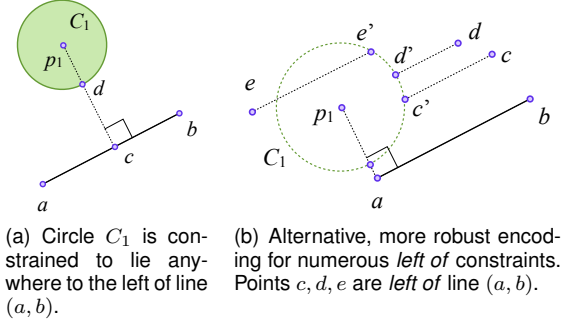


Fig. 6. Relative orientation encodings for regions and more efficient encodings.

D. Topological relations between circles

In standard geometric constraint solvers there is no way of directly specifying mereotopological constraints between higher-level objects and regions such as circles, squares, triangles, polygons, and so on. In this section we present encodings for topological relations between circles, and then use these encodings as a basis for defining relations between more general regions. Firstly we define a useful BRACE relation between a line segment and a circle that ensures the diameter of the circle is equal to the length of the segment (Figure 7(a)).

$$\text{BRACE}(l_{ab}, C_i) \equiv \text{COIN}(l_{ab}, C_i) \wedge \text{COLL}(p_i, l_{ab})$$

We adopt the terminology of the prominent topological spatial logic, the *Region Connection Calculus* (RCC) [30]: *disconnects* (DC), *externally connects* (EC), *partial overlap* (PO), *tangential proper part* (TPP), *non-tangential proper part* (NTPP), *proper part* (PP), *part of* (P), *discrete from* (DR), *equal* (EQ). Note that EQ between two circles is trivially satisfied by constraining the centroids and radii to be equal.

The topological relation encodings are illustrated in Figure 7. To ensure circle intersection (e.g. TPP, NTPP, PO), the encodings constrain one or both endpoints of the brace segments of one circle to be coincident to the brace segment of the other circle; a pair of brace endpoints are made equal for boundary contact (e.g. TPP). EC is encoded with a point of boundary contact a that is coincident to a segment $l_{p_1 p_2}$ between the circle centroids. DC is encoded by introducing a

third circle C_3 so that one endpoint of each of the braces of C_1 and C_2 lie on different sides of the centroid of C_3 , along the brace of C_3 .

Observe that the brace segment within a circle can be rotated about the circle's centroid. Thus, considering NTPP for example, C_1 can occupy any circular region within C_2 by moving C_1 along the brace of C_2 , and rotating the brace of C_2 .

$$\text{TPP}(C_1, C_2) \equiv \exists l_{ab}, l_{ac} \in \mathbf{L} \left(\text{BRACE}(l_{ab}, C_2) \wedge \text{BRACE}(l_{ac}, C_1) \wedge \text{COIN}(c, l_{ab}) \right)$$

$$\text{NTPP}(C_1, C_2) \equiv \exists l_{ab}, l_{cd} \in \mathbf{L} \left(\text{BRACE}(l_{ab}, C_2) \wedge \text{BRACE}(l_{cd}, C_1) \wedge \text{COIN}(l_{cd}, l_{ab}) \right)$$

$$\text{PO}(C_1, C_2) \equiv \exists l_{ab}, l_{cd} \in \mathbf{L} \left(\text{BRACE}(l_{ab}, C_2) \wedge \text{BRACE}(l_{cd}, C_1) \wedge \text{COIN}(a, l_{cd}) \wedge \text{COIN}(d, l_{ab}) \right)$$

$$\text{EC}(C_1, C_2) \equiv \exists a \in \mathbf{P} \left(\text{COIN}(a, l_{p_1 p_2}) \wedge \text{COIN}(a, C_1) \wedge \text{COIN}(a, C_2) \right)$$

$$\text{DC}(C_1, C_2) \equiv \exists a, b \in \mathbf{P}, \exists C_3 \in \mathbf{C} \left(\text{BRACE}(l_{p_1 p_2}, C_3) \wedge \text{COIN}(a, l_{p_1 p_3}) \wedge \text{COIN}(a, C_1) \wedge \text{COIN}(b, l_{p_2 p_3}) \wedge \text{COIN}(b, C_2) \right)$$

We can drop the distinction between boundaries (i.e. corresponding to RCC5 and other RCC relations) by employing the modified coincident constraint between points and segments $\text{COIN}^\subseteq(p, l_{ab})$, where a point p can also equal the segment endpoints l_{ab} . Thus, we encode the definitions that:

- PP is a disjunction of NTPP and TPP;
- P is a disjunction of PP and EQ;
- DR is a disjunction of DC and EC.

$$\text{PP}(C_1, C_2) \equiv \exists l_{ab}, l_{cd} \in \mathbf{L} \left(\text{BRACE}(l_{ab}, C_2) \wedge \text{BRACE}(l_{cd}, C_1) \wedge \text{COIN}^\subseteq(c, l_{ab}) \wedge \text{COIN}^\subseteq(d, l_{ab}) \right)$$

$$\text{P}(C_1, C_2) \equiv \exists l_{ab}, l_{cd} \in \mathbf{L} \left(\text{BRACE}(l_{ab}, C_2) \wedge \text{BRACE}(l_{cd}, C_1) \wedge \text{COIN}^\subseteq(l_{cd}, l_{ab}) \right)$$

$$\text{DR}(C_1, C_2) \equiv \exists a, b \in \mathbf{P}, \exists C_3 \in \mathbf{C} \left(\text{BRACE}(l_{p_1 p_2}, C_3) \wedge \text{COIN}^\subseteq(a, l_{p_1 p_3}) \wedge \text{COIN}^\subseteq(a, C_1) \wedge \text{COIN}^\subseteq(b, l_{p_2 p_3}) \wedge \text{COIN}^\subseteq(b, C_2) \right)$$

E. Egg-yolk approach for defining relations between regions

We employ the egg-yolk method of modelling regions with indeterminate boundaries [11] to characterise a class of regions (including polygons) that satisfies topological and relative orientation relations. Each egg-yolk region is an equivalence class for all regions that are contained within the upper approximation (the *egg white*), and completely contain the lower approximation (the *egg yolk*). Let \mathbf{R} be the domain of

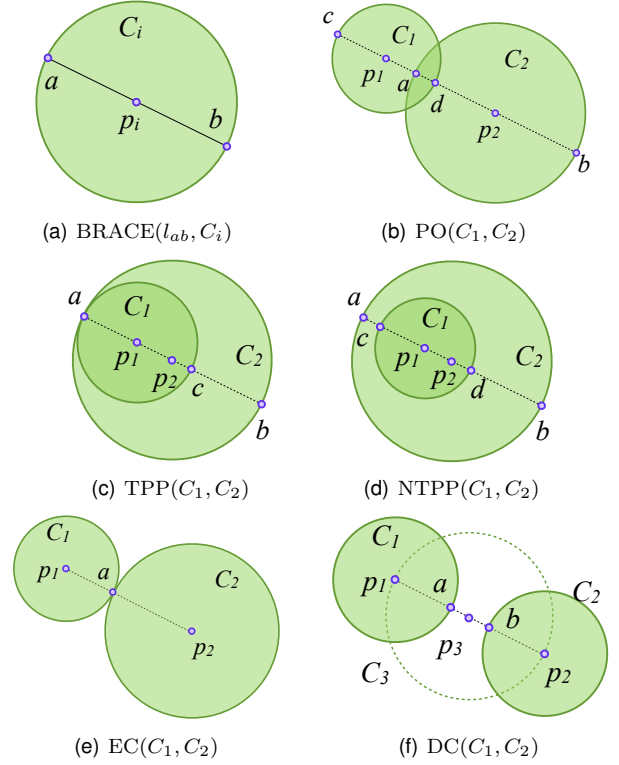


Fig. 7. Topological relations between circles.

egg-yolk regions, where egg-yolk region $R \in \mathbf{R}$ consists of a circular upper and a lower approximation $R^+, R^- \in \mathbf{C}$ such that $\text{NTPP}(R^-, R^+)$ (see Figure 8(a)).

We can realise these regions through constructive geometric constraint encodings, giving us a method of generating arbitrary regions that satisfy qualitative spatial constraints. We declaratively define a (simple, non-self-intersecting) polygon as a sequence of vertices such that:

- 1) all vertices are contained within the upper approximation
- 2) no segment between adjacent vertices intersects the lower approximation
- 3) the (absolute) winding number about the centroid of the lower approximation is 1

We can generate polygons by placing n vertices on the upper approximate circle, evenly distributed (satisfying Condition 3), such that each vertex and line segment is geometrically constrained to satisfy Conditions 1 and 2 above. The user can explore the space of consistent polygons directly through dynamic geometry [33], or polygons can be randomly generated.

Relative orientation between egg-yolk regions and lines can now be defined based on the upper approximations (see Figure 6(a)):

$$\text{LEFT}(R, l_{ab}) \equiv \text{LEFT}(R^+, l_{ab})$$

$$\text{RIGHT}(R, l_{ab}) \equiv \text{RIGHT}(R^+, l_{ab})$$

The following topological relations between pairs of egg-yolk regions are defined based on the relation between their approximations (see Figure 8):

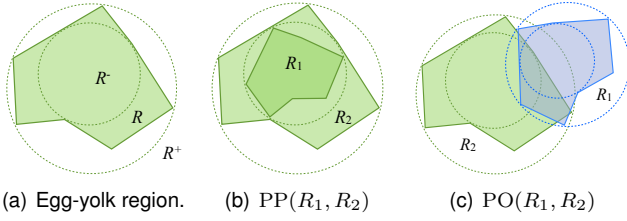


Fig. 8. Egg-yolk region R is defined by a lower circular approximation R^- and an upper circular approximation R^+ .

$$\begin{aligned} PP(R_1, R_2) &\equiv P(R_1^+, R_2^-) \\ DC(R_1, R_2) &\equiv DC(R_1^+, R_2^+) \\ DR(R_1, R_2) &\equiv DR(R_1^+, R_2^+) \\ PO(R_1, R_2) &\equiv PO(R_1^+, R_2^+) \wedge PO(R_1^-, R_2^-) \end{aligned}$$

The *partial overlap* definition requires some explanation: firstly, the partial overlap condition between the upper approximations ensures that the regions each have interior regions not shared by the other, but the regions could still be disconnected. Secondly, the partial overlap condition between the lower approximations ensures that the regions share a common interior region, but one region might completely contain the other. Thus, together the conditions encode the *partial overlap* relation between egg-yolk regions.

The above relative orientation and topological egg-yolk relation encodings are *sound*, i.e. they correctly encode the intended relation between the true regions, and are *incomplete*, i.e. they do not capture all possible ways that the true regions can satisfy the intended relation.

IV. EMPIRICAL EVALUATION

In this section we empirically evaluate our geometric encodings on a range of benchmark problems from the Qualitative Spatial Reasoning community. We have implemented the encodings in CLP(QS) using geometric constraint solvers FreeCAD⁷ and limited BFGS-B [9]. The results show that (a) our system can handle problems from these benchmarks that are unsolvable using relation algebraic methods for qualitative spatial reasoning, and (b) our system outperforms other symbolic algebraic approaches for these benchmark problems by orders of magnitude (such as Cylindrical Algebraic Decomposition). Experiments were run on a MacBookPro, OS X 10.8.5, 2.6 GHz, Intel Core i7.⁸

A. Tent Benchmark Problem

The generalised tent problem [23] is a relative orientation benchmark problem in qualitative spatial reasoning. The problem is specifically designed to be unsolvable using relation algebraic approaches by creating inconsistent relative orientation constraints that cannot be determined to be inconsistent using those methods. Given a set of n distinct 2D points p_1, \dots, p_n , let $LEFT(p_1, p_2, p_3)$ be *true* if p_3 is *left of* the line (p_1, p_2)

⁷ www.freecadweb.org

⁸3D visualisations have been rendered using glc_player: www.glc-player.net

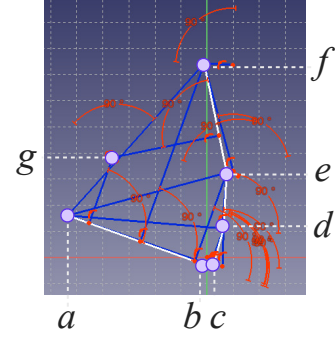


Fig. 9. Consistent tent configuration with $n = 7$ points: a, \dots, g .

where *left of* is interpreted axiomatically (e.g. using relation algebras [26] or a first-order spatial logic [5]) or polynomially (using analytic methods). The consistent tent problem has constraints: $LEFT(p_i, p_j, p_k)$ for all $0 \leq i < j < k \leq n$. The *inconsistent* tent problem has the same constraints with the exception that $LEFT(p_1, p_2, p_n)$, $LEFT(p_2, p_3, p_n)$ are replaced with the equivalent *RIGHT* constraints.

Using an off-the-shelf geometric constraint solver, FreeCAD, our encodings can solve the tent problem for $n = 7$ within 20 seconds for both consistency and inconsistency, which significantly out performs symbolic approaches (in [23] the authors report that quantifier elimination algorithms could not solve for $n = 6$ after 6 hours). We also note that, once the consistent scene has been constructed, incremental updates occur in real-time⁹ e.g. the user can move points around the scene and the FreeCAD solver manipulates the other objects to ensure that the relative orientation constraints are maintained (see Figure 9).

B. Contact problems

A range of contact problems require combining topological and size information that are not solvable using relation algebraic methods. Standard approaches to QSR employ algebraic closure by ensuring that all sub-graphs with 3 vertices are satisfiable. Thus, any problem that inherently requires checking four or more objects simultaneously is beyond algebraic closure. We consider the task of determining the maximum number of spheres and circles that can be mutually externally connected; we also consider the variation where the spheres (circles) must all be the same size.

CLP(QS) correctly solves these spatial contact problems: 5 spheres; 4 same-size spheres; 4 circles; 3 same-size circles (see Figure 10). To demonstrate the scalability of the approach, we consider 100 same size circles (illustrated in Figure 11) and 100 spheres. All of the above contact problems for $n < 10$ are solved within 2 seconds in CLP(QS). Both contact problems with $n = 100$ are solved in 20 seconds.

At each iteration, CLP(QS) produces a series of configurations as it attempts to converge on the solution; an extract of snapshots of the solution history for $n = 100$ same

⁹FreeCAD solver reports solve time of 0.004s.

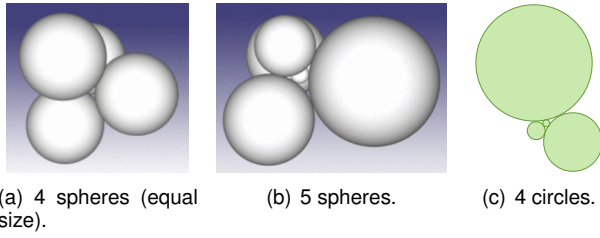
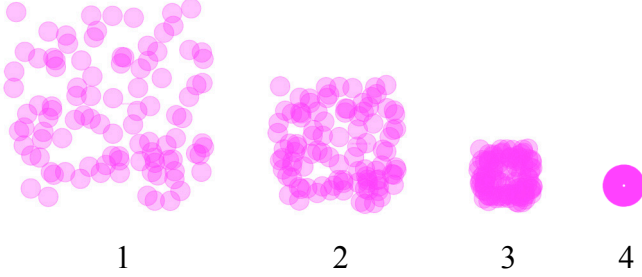


Fig. 10. Contact benchmark problems.

Fig. 11. Four snapshots of the *solving history* of CLP(QS) attempting to solve 100 mutually touching circles, equal size. Snapshot 4 is the best solution found.

size circles is illustrated in Figure 11. By treating time as a third dimension we can construct a *space-time history* region [17, 18] from the sequence of intermediate configurations as illustrated in Figure 12. We can directly visualise and study the qualitative relations between these history regions to provide further insight into the nature of the problem at hand and the underlying solving process, e.g. automatically generating natural language explanations of the search space explored during the solving process.

C. Computer Aided Product Design

The task is to design an adjustable desk lamp with the following qualitative requirements: the lamp has a base and three bars connected by three joints; the joints can only turn *inwards*; the lamp shade connects to the third joint; the bulb must fit completely within the lamp shade. Figure 13 illustrates the constraint graph and corresponding FreeCAD interactive diagram that maintains the specified qualitative relations. As the user manipulates the diagram, the FreeCAD geometric solver maintains the qualitative constraints in real time (reported solving time: 0.001 seconds).

V. DISCUSSION AND CONCLUSIONS

We have presented novel geometric encodings of qualitative relations for: relative orientation between lines and either points or regions; mereotopological relations between circles (or, more accurately, disks); topological relations between regions represented by an upper and lower approximation (i.e. providing an approach for reasoning about polygons and regions bounded by jordan curves).

Importantly, our encodings are based on a standard geometric constraint language that is well known within the

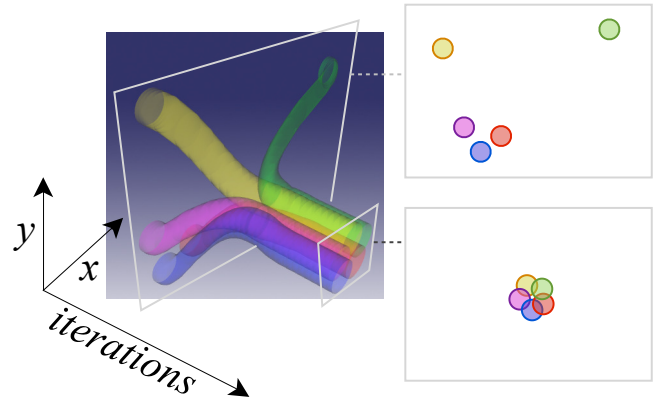
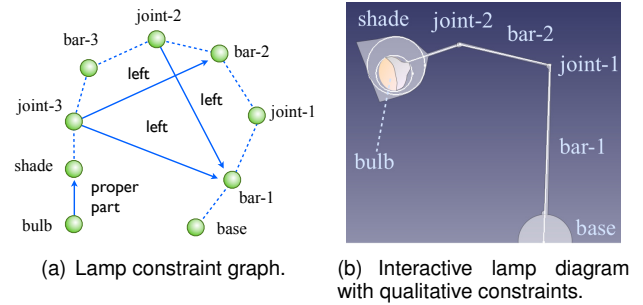
Fig. 12. *Space-time history* volumes derived from intermediate configurations produced during the solving process. Task requirement is five mutually touching, same-sized circles.

Fig. 13. Lamp shade product design with qualitative constraints.

constructive geometric constraint solving community: points, lines, and circles, with relations that correspond to *quadratic equations* (parallel, coincidence, perpendicularity, dimension constraints, etc.). This standard language is adopted by all prominent solvers that we are aware of, particularly in the Computer Aided Design and Manufacturing domain. Thus, our encodings can be directly employed in *all* prominent commercially available constructive geometric constraint solvers (including Autodesk Inventor, FreeCAD, LEDAS LGS2D) to extend those systems to also reason about qualitative spatial relations.

Constructive geometric constraint solving is significantly more computationally efficient compared to other approaches for solving systems of polynomial constraints (e.g. Cylindrical Algebraic Decomposition, the Gröbner basis method, and Wu's characteristic set method all have double exponential complexity), and can scale to real-world problems that involve hundreds of spatial objects. However, the supported spatial language is rather restricted, only permitting relations that can be encoded as quadratic equations. Traditionally, qualitative orientation and mereotopological relations are encoded using *inequalities*, thus ruling out the use of standard geometric constraint solvers; directly encoding inequalities for numerical optimisation methods is also known to be significantly less robust compared to quadratic equations, as it introduces

saddles near optima.

Our key contribution is showing that indeed we can encode these qualitative relations based on the restricted standard geometric constraint language. Thus, we have an efficient method for solving problems involving qualitative orientation and mereotopology beyond computationally intractable symbolic methods. The results of our empirical evaluation, based on an implementation in FreeCAD and limited BFGS-B (providing numerical optimisation, which is effective for solving certain sub-problems), show that our encodings can solve contact and orientation benchmark problems within seconds that take hours (or more) using other symbolic approaches such as Cylindrical Algebraic Decomposition.

One open problem is determining the most efficient encodings for certain problem classes. Our encodings are certainly not unique, and many alternative encodings can be employed that have different properties, e.g. comparing encodings that minimise the number of objects and constraints introduced as the problem size increases, or encodings that avoid the use of computationally more expensive constructions (i.e. encodings that introduce more objects and constraints, but simpler constraints to solve for). We are continuing to identify more efficient encodings to further increase the horizon of solvable real-world problems, and benchmark problems, within the context of declarative constraint logic programming over qualitative spatial domains.

Another interesting open question is how to handle inconsistent qualitative constraints in general: methods such as Cylindrical Algebraic Decomposition are both sound and complete, whereas constructive geometric constraint solving is incomplete in general. Thus, a result of inconsistency using constructive approaches is usually annotated with some measure of confidence (i.e. the problem, or sub-problems, are executed a number of times with different initial randomised parameter values until no further progress towards a solution is made). Identifying tractable classes of qualitative problems that have specific properties with respect to completeness (and statistical confidence in the case of reported inconsistency) is an interesting direction for future research.

VI. ACKNOWLEDGEMENTS

We sincerely thank the reviewers for their very careful reading of the paper, particularly in their diligence in reviewing our definitions. This work is funded by the German Research Foundation (DFG) under grant for a SketchMapia project (Grant SCHW 1372/7-1).

REFERENCES

- [1] M. Aiello, I. E. Pratt-Hartmann, and J. F. v. Benthem. *Handbook of Spatial Logics*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2007. ISBN 978-1-4020-5586-7.
- [2] D. S. Arnon, G. E. Collins, and S. McCallum. Cylindrical Algebraic Decomposition I: The basic algorithm. *SIAM Journal on Computing*, 13(4):865–877, 1984.
- [3] M. Bhatt, J. H. Lee, and C. Schultz. CLP(QS): A Declarative Spatial Reasoning Framework. In *Proceedings of the 10th international conference on Spatial information theory*, COSIT’11, pages 210–230, Berlin, Heidelberg, 2011. Springer-Verlag. ISBN 978-3-642-23195-7.
- [4] M. Bhatt, C. Schultz, and C. Freksa. The ‘Space’ in Spatial Assistance Systems: Conception, Formalisation and Computation. In T. Tenbrink, J. Wiener, and C. Claramunt, editors, *Representing space in cognition: Interrelations of behavior, language, and formal models. Series: Explorations in Language and Space*. 978-0-19-967991-1, Oxford University Press, 2013.
- [5] S. Borgo. Euclidean and mereological qualitative spaces: A study of SCC and DCC. In C. Boutilier, editor, *IJCAI 2009, Proceedings of the 21st International Joint Conference on Artificial Intelligence, Pasadena, California, USA, July 11-17, 2009*, pages 708–713, 2009. URL <http://ijcai.org/papers09/Papers/IJCAI09-123.pdf>.
- [6] D. Bouhineau. Solving geometrical constraint systems using CLP based on linear constraint solver. In *Artificial Intelligence and Symbolic Mathematical Computation*, pages 274–288. Springer, 1996.
- [7] D. Bouhineau, L. Trilling, and J. Cohen. An application of CLP: Checking the correctness of theorems in geometry. *Constraints*, 4(4):383–405, 1999.
- [8] W. Bouma, I. Fudos, C. Hoffmann, J. Cai, and R. Paige. Geometric constraint solver. *Computer-Aided Design*, 27(6):487–501, 1995.
- [9] R. H. Byrd, P. Lu, J. Nocedal, and C. Zhu. A limited memory algorithm for bound constrained optimization. *SIAM Journal on Scientific Computing*, 16(5):1190–1208, 1995.
- [10] S.-C. Chou. *Mechanical geometry theorem proving*, volume 41. Springer Science & Business Media, 1988.
- [11] A. G. Cohn and N. M. Gotts. The Öegg-yolkÖrepresentation of regions with indeterminate boundaries. *Geographic objects with indeterminate boundaries*, 2: 171–187, 1996.
- [12] G. E. Collins. Quantifier elimination for real closed fields by cylindrical algebraic decomposition. In *Automata Theory and Formal Languages 2nd GI Conference Kaiserslautern, May 20–23, 1975*, pages 134–183. Springer, 1975.
- [13] G. E. Collins and H. Hong. Partial cylindrical algebraic decomposition for quantifier elimination. *Journal of Symbolic Computation*, 12(3):299 – 328, 1991. ISSN 0747-7171. doi: [http://dx.doi.org/10.1016/S0747-7171\(08\)80152-6](http://dx.doi.org/10.1016/S0747-7171(08)80152-6). URL <http://www.sciencedirect.com/science/article/pii/S0747717108801526>.
- [14] X.-S. Gao and S.-C. Chou. Solving geometric constraint systems. ii. a symbolic approach and decision of rc-constructibility. *Computer-Aided Design*, 30(2):115–122, 1998.
- [15] X.-S. Gao and S.-C. Chou. Solving geometric constraint systems. i. a global propagation approach. *Computer-Aided Design*, 30(1):47–54, 1998.
- [16] J.-X. Ge, S.-C. Chou, and X.-S. Gao. Geometric constraint satisfaction using optimization methods. *Computer-Aided Design*, 31(14):867–879, 1999.
- [17] P. J. Hayes. The second naive physics manifesto. In J. R. Hubbs and R. C. Moore, editors, *Formal Theories of the Commonsense World*. Ablex Publishing Corporation, Norwood, NJ, 1985.
- [18] S. M. Hazarika. *Qualitative Spatial Change : Space-Time Histories and Continuity*. PhD thesis, The University of Leeds, School of Computing, 2005. Supervisor - Anthony Cohn.
- [19] T. L. Heath (ed). *The thirteen books of Euclid’s Elements*, volume 1. Courier Dover Publications, 1956.
- [20] D. Kapur and J. L. Mundy, editors. *Geometric Reasoning*. MIT Press, Cambridge, MA, USA, 1988. ISBN 0-262-61058-2.
- [21] P. B. Ladkin and R. D. Maddux. On binary constraint problems. *Journal of the ACM (JACM)*, 41(3):435–469, 1994.
- [22] J. H. Lee. The complexity of reasoning with relative directions. In *21st European Conference on Artificial Intelligence (ECAI 2014)*, 2014.
- [23] J. H. Lee and D. Wolter. A new perspective on reasoning with qualitative spatial knowledge. In *IJCAI-2011 Workshop 27*, page 3, 2011.
- [24] Y.-T. Li, S.-M. Hu, and J.-G. Sun. A constructive approach to solving 3-d geometric constraint systems using dependence analysis. *Computer-Aided Design*, 34(2):97–108, 2002.
- [25] G. Ligozat. *Qualitative Spatial and Temporal Reasoning*. Wiley-ISTE, 2011.
- [26] G. F. Ligozat. Qualitative triangulation for spatial reasoning. In *Spatial Information Theory A Theoretical Basis for GIS*, pages 54–68. Springer, 1993.
- [27] J. C. Owen. Algebraic solution for geometry from dimensional constraints. In *Proceedings of the first ACM symposium on Solid modeling foundations and CAD/CAM applications*, pages 397–407. ACM, 1991.
- [28] G. Pesant and M. Boyer. QUAD-CLP (R): Adding the power of quadratic constraints. In *Principles and Practice of Constraint Programming*, pages 95–108. Springer, 1994.
- [29] G. Pesant and M. Boyer. Reasoning about solids using constraint logic programming. *Journal of Automated Reasoning*, 22(3):241–262, 1999.
- [30] D. A. Randell, Z. Cui, and A. Cohn. A spatial logic based on regions and connection. In *KR’92. Principles of Knowledge Representation and Reasoning*, pages 165–176. Morgan Kaufmann, San Mateo, California, 1992.
- [31] C. Schultz and M. Bhatt. Towards a Declarative Spatial Reasoning System. In *20th European Conference on Artificial Intelligence (ECAI 2012)*, 2012.
- [32] C. Schultz and M. Bhatt. Declarative spatial reasoning with boolean combinations of axis-aligned rectangular polytopes. In *ECAI 2014 - 21st European Conference on Artificial Intelligence*, pages 795–800, 2014.
- [33] H. Winroth. *Dynamic projective geometry*. Tekniska högsk., 1999.

Using Mathematical Modeling as an Example of Qualitative Reasoning in Metaphysics. A Note on a Defense of the Theory of Ideas

Bartłomiej Skowron

International Center for Formal Ontology

Department of Administration and Social Sciences, Warsaw University of Technology

ul. Plac Politechniki 1, 00-661 Warszawa, Poland

Pontifical University of John Paul II in Cracow

Email: bartlomiej.skowron@gmail.com

<http://www.math.uni.wroc.pl/~s160571/nowa.html>

Abstract—This is an outline of a defense of the theory of Ideas. I propose—using qualitative reasoning in metaphysics—the new incarnation of the theory of Ideas and I try to defend the theory against traditional counterarguments. The starting point are the theories of Ideas of Plato and Ingarden and an ontology of Ideas proposed by Kaczmarek; these theories are paraphrased—using a modified method of semantic paraphrases of Ajdukiewicz—and presented in terms of the basic concepts of category theory. To paraphrase Ideas as categories I propose recognized category theory as a pattern for the theory of Ideas. This recognition—based on an *analogy* between mathematical structures and philosophical structures—is the core of the qualitative reasoning in metaphysics. It could also be called a mathematical philosophy or mathematical modeling in metaphysics. I invoke an arrows-like, i.e. no-object-oriented, formulation of a category and I base the proposed theory of Ideas on that formulation. The components of an Idea are arrows and their compositions (equivalents of changes and transformations); objects in this approach are special arrows namely the identity arrows. Using the category of higher dimensions I introduce the concept of the dimension of an Idea (and other concepts) which allows me to refute the argument of the “third man”.

I. BACKGROUND

PLATO divided reality into what really exists (*Forms* or *Ideas*) and everything else. The first is not easily perceptible, the second is indeed perceptible and tangible, but essentially it is not real—it just reflects the first. Plato’s Ideas were a perfect and ideal realm, they were eternal and changeless. Things that appear to our senses only participate in the Ideas, imitate them; Ideas are models for things that are present in things. The realm of things is variable and unstable, they arise and perish, Ideas are timeless, constant, stable, independent and original. One can only learn about them intellectually. It is beyond a doubt that the Platonic bifurcation of reality is of fundamental importance to Western thought. According to A. N. Whitehead: “The safest general characterization of the European philosophical tradition is that it consists of a series of footnotes to Plato” (Whitehead, 1929, p. 39).

Aristotle pointed out many difficulties in Plato’s theory of Ideas, including the one that there are Ideas of negation (Negation Argument), but there should not be; he pointed to the argument of the “third man”. As he claimed: “[T]o say that they are patterns and the other things share in them is to use empty words and poetical metaphors” (*Met.* I, 997b 5–12). He also complained of his contemporaries that mathematics “has come to be identical with philosophy”. Next Aristotle inquired as to how many Ideas can arise from one (One-Many Argument), as is the case with numbers. With dissatisfaction and disbelief he accepted the fact that Ideas can be patterns not only for things, but also for other Ideas (Pattern Argument).

A. Prior Contemporary Research on Ideas

Philosophy of the twentieth century, especially phenomenology and analytic philosophy, has undertaken the task of creating new theories of Ideas. Some of the works of Roman Ingarden (Ingarden, 1925) and Jean Héring (Héring, 1921) on the one hand, and Kit Fine (Fine, 1995), Edward Zalta (Zalta, 1983) and Janusz Kaczmarek (Kaczmarek, 2008) on the other hand were devoted, in a certain way, to Ideas.

Ingarden and Kaczmarek’s study of the Ideas most influenced a formulation of the proposed dynamic and structural theory of Ideas. Ingarden in his *Essentiale Fragen* partially modifying the account of Plato and partially rejecting it, presented a full-fledged theory of Ideas. Ingarden captured the formal two-sidedness of Ideas and introduced constants and variables in the content of Ideas. He examined and analyzed the ontological source of the content of Ideas, that is to say the *pure ideal qualities*. In Kaczmarek’s account the ideal objects were represented by a pair of functions. From the point of view of the theory of Ideas presented in this note the crucial point of Kaczmarek’s formalized ontology is that the objects were represented by functions, not by elements or ingredients. The “material” of ideal objects were functions.

A further stimulus of the proposed theory was Thomas Mormann Theory of Object. Mormann (Mormann, 1997)—inspired by David Armstrong (Armstrong, 1989) and Brian

Skyrms (Skyrms, 1993)—presented an ontology of objects as functions. More specifically he has created a combinatorial theory of possible worlds using the concept of mapping. The possible world in his account is a continuous mapping from the individuals to the space of properties. As in Kaczmarek's account the underlying "mathematical materials" were funtions.¹

B. An Outline of the Note

The aim is to defend the theory of Ideas against traditional counterarguments. The starting point are the theories of Ideas of Plato and Ingarden and an ontology of Ideas proposed by Kaczmarek; these theories are paraphrased—using a modified method of semantic paraphrases of Ajdukiewicz—and presented in terms of the basic concepts of category theory. To paraphrase Ideas as categories I propose recognized category theory as a pattern for the theory of Ideas. I invoke an arrows-like, i.e. no-object-oriented, formulation of a category and I base the proposed theory of Ideas on that formulation. The components of an Idea are arrows and their compositions (equivalents of changes and transformations); objects in this approach are special arrows (identity arrows). One can understand the arrow as it appears in category theory as a generalized function—in this sense our approach is the generalization of the above-mentioned Kaczmarek and Mormann's study. Using the category of higher dimensions I introduce the concept of dimension of an Idea (and other concepts) which allows me to refute the argument of the "third man".

II. MATHEMATICAL PHILOSOPHY

A. Ajdukiewicz's Semantic Paraphrase of the Philosophical Issue as an Example of Qualitative Reasoning in Philosophy

In general we can say that the semantic paraphrase method is the method of translating traditional philosophical issues into the language of logic (or metalogic). The method consists of the following steps cf. (Woleński, 1989):

- (i) the most accurate and clear identification of the philosophical issues under consideration;
- (ii) selection of a logical theorem having a similar structure to that formulated problem of (i);
- (iii) determination of certain compounds (of syntactic, semantic or pragmatic nature) between the expressions of (i) and the expressions of a logical theorem from (ii);
- (iv) construction of a paraphrase, that is a sentence of a "isomorphic" structure with a selected logical theorem, i.e. an unambiguous assignment of meanings of expressions from (i) meanings of expressions from (ii);
- (v) to validation of the paraphrase by
 - (v_a) phenomenological analysis,
 - (v_b) meaning postulates, i.e. assigning in an arbitrary manner some meaning to the expres-

sions; by convention, or postulates or some definitions;

- (vi) drawing consequences from the paraphrase;
- (vii) evaluation of these consequences from the perspective of the philosophical problem under consideration.

Ajdukiewicz in *A Semantical Version of the Problem of Transcendental Idealism* (Ajdukiewicz, 1977, p. 140–154) analysed the doctrine of transcendental idealism which amounts to the claim that "reality is a correlate of the transcendental subject" (see also (Przełęcki, 1990)). This thesis was translated into the claim that any statement of the language of science is true if and only if it is "dictated by the meaning-rules of that language". In fact, from a metalogical point of view it is obvious that this position is not correct.

The core of Ajdukiewicz's method is the above-mentioned translation (or construction of a paraphrase) which must be preceded by an appropriate recognition of the problem.

B. A Revised Version of Ajdukiewicz's Method

Ajdukiewicz was a logic-centered philosopher. For this reason the second step (ii) of his method consists only of logical theorems. I am elaborating his method by extending its applicability to the entire field of mathematics—not only to logic or metalogic.² Furthermore Ajdukiewicz believed that we would find similarities between theorems that are in fact between language expressions. I suggest we focus our attention on the analogy between the formal structures hidden behind the problem in question. It is not easy to say what these underlying formal structures are.³ Indeed the proper answer to this question requires the general metaphysical theory of formal structures. Mac Lane puts forward one of the possible theories of the origins and development of the structures in his *Mathematics: Form and Function* (Mac Lane, 1986). Mac Lane as a philosopher was anti-platonic, nevertheless his ideas can be used to explain the concept of the structure and the analogy between the structures even in our platonic approach.

C. Basic Paraphrase: Ideas as Categories

I suggest—thinking qualitatively—that the categories of category theory correspond to the Ideas. This correspondence is the (iv) step of the paraphrase method described above. The Ideas are composed of similar data as the categories and behave similarly to the categories. In other words, I suggest that category theory is a good approximation (a model? a material?) and description of the theory of Ideas.

In order to justify the paraphrase, to satisfy the (v) condition from Ajdukiewicz's method, one should say the following: categories (in terms of category theory) are similar to Ideas (in terms of philosophy) in many respects; objects are incarnations of Ideas, fall within Ideas, participate in them, and—mimicking them—are modelled by them in metaphysics.

²An ontological analysis of the extension carried out using the topological tools can be found here (Skowron, 2014).

³In this sense Hilbert spaces in quantum mechanics are the underlying structures of the quantum world.

¹Needless to say that Mormann's account has nothing to do with Ideas.

Objects of mathematics, like groups, rings, fields, orders, proofs and so on, also fall within categories, in the sense of taking part in them, or are modelled by them⁴. Category theory is a theory of Ideas for mathematical objects. Not only mathematical objects are involved in categories. An application of category theory (and also all mathematics) in modern physics testifies to the fact that categories are in some way also present in the real world.

Ideas are the categories. Using the ontological analysis of Ideas carried out by Ingarden (Ingarden, 1925, 1965) one can say that a constant of the content of Ideas is based upon the fact that we are talking about this and not another category. A constant of the group category **GRP** will therefore be the fact that its objects are groups and its arrows are the group homomorphisms. The variable content of the category of groups **GRP** is the possibility of concretization of a variety of groups. The constant of the group category **GRP** is an ideal concretization of the possibility of concretization of some groups and their homomorphisms.

III. THE METAPHYSICAL RESULT OF QUALITATIVE REASONING

Philosophers such as Plato and Ingarden focused on describing the interior of Ideas. They believed that the essence of Ideas is hidden inside them. In fact, the opposite is the case. Ideas are byproducts of *change*. It is convenient to use the expression *Idea*, but the essence of Ideas is that they are secondary entities with respect to changes and transformations. That is why one should focus on the relationships between Ideas.

1) *Why All This? The Newly Identified Features of Ideas:*

On the basis of paraphrases one can identify the new features of Ideas:

- 1) the dynamical structure of Ideas (referring to the arrows-only definition of a category (Mac Lane, 1998, p. 9));
- 2) the object of an Idea as the identity transformation within the Idea;
- 3) the dimension of Ideas (referring to the n -category theory (Baez, 1997));
- 4) the negations and oppositions of Ideas (referring to the concept of dual category C^{op});
- 5) the *start* and *end* of some Ideas (referring to the initial and terminal objects);
- 6) the full-fledged structural and, in fact, *transcendental* mereology of Ideas (referring to the proposal of updating mereology by a category theory done by Mormann (Mormann, 2009));
- 7) the tangled Ideas (on the basis of the adjoint functors (Mac Lane, 1998, p. 79–108));
- 8) the Idea of the Ideas (referring to the category of small categories);
- 9) the problem of the spatiality of Ideas (referring to the geometrical aspects of categories).

⁴It is worth mentioning that the Ideas in metaphysics can also fall within Ideas, not only within real objects. The Idea of man falls within the Idea of animal.

A. *Refutation of the Arguments Against the Theory of Ideas*

Using the identified properties of Ideas I can reject the following arguments:

- the Third Man Argument (by using the dimensions of Ideas)
- the Negation Argument (by using the opposite category)
- the One-Many Argument (by using the basic properties of the creation of a new category from an old one)
- the Pattern Argument (following the basic properties of n -dimensional Ideas)

B. *Ideas are Byproducts of Change*

The characteristics of a category introduces three types of entities: objects, arrows and compositions. Categories, however, were introduced in order to define the morphisms (and morphisms to define a natural transformation), namely to give them domain and counterdomain (Eilenberg and Mac Lane, 1945, p. 247). The concept of category was therefore an auxiliary concept. This is expressed in the fact that a category could be equivalently defined without the concept of object and using the concept of the arrow and composition (Mac Lane, 1998, p. 9)

On the basis of the paraphrase method, to be more specific by the condition number (vi), one can claim that an Idea is just a byproduct of transformations or a pure change. The realm of Ideas is not as stable and unchangeable as Plato imagined. In the face of the fact that a structure of the category **TOP** is not known in all its details, one can claim that these totalities of Ideas are complicated and there are no easy and obvious instruments to comprehend them. The ontological structure of the realm is a network of mutually influencing and co-creating Ideas.

REFERENCES

- Ajdukiewicz, K. (1977). *The Scientific World-Perspective and Other Essays, 1931-1963*. Synthese Library. Springer.
- Armstrong, D. (1989). *A Combinatorial Theory of Possibility*. Cambridge Studies in Philosophy. Cambridge University Press.
- Awodey, S. (1996). Structure in Mathematics and Logic: A Categorical Perspective. *Philosophia Mathematica*, 4(3):209–237.
- Awodey, S. (2014). Structuralism, Invariance, and Univalence. *Philosophia Mathematica*, 22(1):1–11.
- Baez, J. C. (1997). An introduction to n -categories. In Moggi, E. and Rosolini, G., editors, *Category Theory and Computer Science*, volume 1290 of *Lecture Notes in Computer Science*, pages 1–33. Springer Berlin Heidelberg.
- Beck, M. (1929). Ideelle Existenz. *Philosophische Hefte*, pages 151–239.
- Bell, J. L. (2001). Observations on category theory. *Axiomathes*, 12(1-2):151–155.
- Eilenberg, S. and Mac Lane, S. (1945). General theory of natural equivalences. *Trans. Am. Math. Soc.*, 58:231–294.
- Fine, K. (1995). The logic of essence. *Journal of Philosophical Logic*, 24(3):241–273.
- Héring, J. (1921). Bemerkungen über das Wesen, die Wesenheit und die Idee. *Jahrbuch für Philosophie und phänomenologische Forschung*, pages 497–543.
- Ingarden, R. (1925). Essentielle Fragen. ein Beitrag zu dem Wesensproblem. (eng. Essential Questions; pol. O pytaniach esencjalnych). *Jahrbuch für Philosophie und phänomenologische Forschung*, pages 125–304.
- Ingarden, R. (1965). *Der Streit um die Existenz der Welt* (Eng. *Controversy over the Existence of the World*; Pol. *Spór o istnienie świata*). M. Niemeyer.
- Kaczmarek, J. (2008). *Indywidualna. Idee. Pojęcia. Badania z zakresu ontologii sformalizowanej (Individuals. Ideas. Concepts. Research on formal ontology)*. Wydawnictwo Uniwersytetu Łódzkiego, Łódź.

- Lawvere, F. W. (1969). Adjointness in foundations. *Dialectica*, 23(3-4):281–296.
- Mac Lane, S. (1986). *Mathematics: Form and Function*. Springer-Verlag, New York, Berlin, Heidelberg, Tokyo.
- Mac Lane, S. (1998). *Categories for the Working Mathematician*. Graduate Texts in Mathematics. Springer.
- Mormann, T. (1997). Topological aspects of combinatorial possibility. *Logic and Logical Philosophy*, 5:75–92.
- Mormann, T. (2009). Updating classical mereology. In Glymour, C., Westerstahl, D., and Wang, W., editors, *Logic, Methodology and Philosophy of Science. Proceedings of the 13th International Congress*. King's College.
- Plato (1997). *Plato. Complete Works*. Hackett Pub.
- Przełęcki, M. (1990). Semantic reasons for ontological statements: The argumentation of a reist. In Woleński, J., editor, *Kotarbiński: Logic, Semantics and Ontology*, volume 40 of *Nijhoff International Philosophy Series*, pages 85–96. Springer Netherlands.
- Simmons, H. (2011). *An Introduction to Category Theory*. Cambridge University Press.
- Skowron, B. (2014). The Forms of Extension. In Szatkowski, M. and Rosiak, M., editors, *Substantiality and Causality*, pages 191–203. Walter de Gruyter, Berlin/New York.
- Skyrms, B. (1993). Logical atoms and combinatorial possibility. *Journal of Philosophy*, 60(5):219–232.
- Whitehead, A. (2010 (1929)). *Process and Reality*. Free Press.
- Woleński, J. (1988). *Logic and Philosophy in the Lvov–Warsaw School*. Synthese Library. Springer Netherlands.
- Woleński, J. (1989). Kierunki i metody filozofii analitycznej. In Perzanowski, J., editor, *Jak filozofować, Studia z metodologii filozofii*, pages 30–77. Polskie Wydawnictwo Naukowe.
- Zalta, E. (1983). *Abstract Objects: An Introduction to Axiomatic Metaphysics*. Synthese Library. Springer.

