

Reasoning with Mixed Qualitative-Quantitative Representations of Spatial Knowledge

Giorgio De Felice

Dissertation
zur Erlangung des Grades eines Doktors der
Ingenieurwissenschaften
— Dr.-Ing. —

Vorgelegt im Fachbereich 3 (Mathematik & Informatik)
der Universität Bremen
im Juni 2012

Datum des Promotionskolloquiums: 26 Juli 2012

Gutachter: Prof. Christian Freksa, Ph.D. (Universität Bremen)
Assoc. Prof. Dr. Chris S. Renschler (University at Buffalo)

Abstract

Drastic transformations in human settlements are caused by extreme events. As a consequence, descriptions of an environment struck by an extreme event, based on spatial data collected before the event, become suddenly unreliable. On the other hand, time critical actions taken for responding to extreme events require up-to-date spatial information. Traditional methods for spatial data collection are not able to provide updated information rapidly enough, calling for the development of new data collection methods.

Reports provided by actors involved in the response operations can be considered as an alternative source of spatial information. Indeed, reports often convey spatial descriptions of the environment. The extraction of spatial descriptions from such reports can serve a fundamental role to update existing information which is usually maintained within, and by means of, Geographic Information Systems. However, spatial information conveyed by human reports has *qualitative* characteristics, that strongly differ from the *quantitative* nature of spatial information stored in Geographic Information Systems. Methodologies for integrating qualitative and quantitative spatial information are required in order to exploit human reports for updating existing descriptions of spatial knowledge.

Although a significant amount of research has been carried on how to represent and reason on qualitative data and qualitative information, relatively little work exists on developing techniques to combine the different methodologies. The work presented in this thesis extends previous works by introducing a hybrid reasoning system—able to deal with mixed qualitative-quantitative representations of spatial knowledge—combining techniques developed separately for qualitative spatial reasoning and quantitative data analysis.

The system produces descriptions of the spatial extent of those entities that have been modified by the event (such as collapsed buildings), or that were not existing before the event (such as fire or ash clouds). Furthermore, qualitative descriptions are produced for all entities in the environment. The former descriptions allow for overlaying on a map the information interpreted

from human reports, while the latter triggers warning messages to people involved in decision making operations. Three main system functionalities are investigated in this work: The first allows for translating qualitative information into quantitative descriptions. The second aims at translating quantitative information into qualitative relations. Finally, the third allows for performing inference operations with information given partly qualitatively and partly quantitatively for boosting the spatial knowledge the system is able to produce.

for the memory of my grandfather

Acknowledgements

Many people supported me in many different ways during my years as a doctoral student and I am thankful to all of them.

First of all, I express my sincere gratitude to my advisor Christian Freksa. I am grateful for giving me the necessary freedom to pursue my ideas and for his support at every stage of my thesis. I want to thank Chris Renschler for the inspiring discussions in Buffalo and for being my second supervisor.

I gratefully acknowledge the financial support provided by the the Deutsche Forschungsgemeinschaft (DFG) under grant IRTG GRK 1498 Semantic Integration of Geospatial Information.

I want to thank all my colleagues at the IRTG group in Bremen for the continuous support they gave me with regard to my work and beyond the scope of this dissertation. Jan Oliver Wallgrün, Falko Schmid, Paolo Fogliaroni, Torben Gerkenmeyer, Manfred Eppe, and Jae Hee Lee, you all deserve my gratitude. In particular, Paolo has been more than a colleague. I would not have started this dissertation without his persuasive arguments.

I want to thank all my colleagues at the Cognitive Systems group for the cozy atmosphere. In particular, I would like to thank Mehul Bhatt and Carl Schultz for being available at any time for comments and support. My sincere thanks go to Lutz Frommberger, Falko Schmid, Holger Schultheis, Carl Schultz, and Diedrich Wolter for volunteering to proofread part of this dissertation.

This dissertation would have been impossible without the continuous and endless support of my family. I want to thank my parents Concetta and Dante, my sister Silvia, my brother Gianfranco, and my grandparents Pierina and Filippo. I dedicate this dissertation to you all.

Richard Bach once wrote, *“There’s no such place as far away”*. Thank you Martina for continuously proving me that it is not just a tale, and for being the best friend I can think of.

I can not conclude my acknowledgements without mentioning all those people that have been a continuous presence in my every day life. I am thankful

to Dino, Federica, Nene, Chicco and Momo. I express my sincere gratitude to Piggi “Bold Wife”, Claù “Irritating Fringe”, Pignateaux “Hippie”, Darilla “Arabic Dwarf”, Giulietta “Little Bonnie”, Paolo “Uncle Paulo”, Matteo “Melted”, Sergio “Little Mate”, Giuseppe “Little Peppe”, Jihad “Jeans”, Michela, Gianluca, and Chiara for being constant reference points in my life. I had the honor to meet lot of inspiring people during my years as a doctoral student, both in Bremen and in Buffalo. I am delighted to thank Mitja, Jesus, Dayana, Jorge, Aneta and Lena, who have been able to play a fundamental role in my life. Last but not least, my best thank goes to Herr Von Fröhligkeit. I apologize if I forgot to mention somebody that surely deserves my sincere gratitude: I dedicate Daniel Pennac’s novel “*Merci*” to them all.

Contents

List of Figures	ix
List of Tables	xiii
1 Introduction	1
1.1 Motivation	1
1.2 Problem Characterization	4
1.2.1 Geographic Information Systems	4
1.2.2 Geographic Information After Extreme Events	5
1.2.3 Information Integration After Extreme Events	6
1.3 Thesis and Contribution	7
1.4 Outline	8
2 Geographic Information Manipulation	11
2.1 Quantitative Representations and Computational Geometry	11
2.1.1 Quantitative Representations	12
2.1.2 Computational Geometry Algorithms	15
2.1.3 Data Collection Methods	18
2.2 Qualitative Spatial Representation and Reasoning	20
2.2.1 Qualitative Spatial Calculi	22
2.2.2 Representation of Spatial Configurations	27
2.2.3 Topology, Cardinal Directions, and Visibility Calculi	29
2.2.4 On the Combination of Qualitative Calculi	38
2.2.5 Spatial Information Translation	40
2.3 Uncertain Spatial Knowledge	41
2.3.1 Modeling Uncertain Spatial Knowledge	42
2.3.2 Uncertain Spatial Knowledge in GIS	43
2.3.3 Qualitative Relations Between Uncertain Regions	44

3	Spatial Information Integration for EM Response	45
3.1	Geographic Data Collection for Emergency Response	45
3.1.1	Emergency Management	46
3.1.2	Example: Information Availability after the Haiti Earthquake . .	48
3.1.3	Information Lack in the EM Response Phase	51
3.1.4	VGI for Emergency Management	51
3.2	A Geographic Information Integration System	52
3.2.1	Spatial Information Extraction and Representation	53
3.2.2	Spatial Information Integration	54
3.2.3	System Description	56
3.3	Qualitative and Quantitative Spatial Information Integration	57
3.3.1	Quantification	57
3.3.2	Spatial Regions with Infinite Extent	59
3.3.3	Imprecise Description of Spatial Regions	60
3.3.4	Quantification with Imprecise Reference Objects	64
3.3.5	Qualification	65
3.3.6	Reasoning	66
3.4	Summary	68
4	Quantification of Qualitative Spatial Information	69
4.1	A System for Spatial Information Translation	70
4.2	Quantification Challenges	71
4.3	Representation of Regions with Infinite Extent	73
4.4	Spatial-Region Objects	74
4.5	The Quantification Component	75
4.5.1	Quantification Algorithm	75
4.6	Quantification of Qualitative Spatial Relations	76
4.6.1	<i>Quantify_C</i> for Cardinal Direction Relations	76
4.6.2	<i>Quantify_C</i> for Visibility Relations	85
4.6.3	<i>Quantify_C</i> for Topological Relations	94
4.7	Computational Complexity of Quantification	97
4.7.1	Cardinal Direction Quantification	98
4.7.2	Visibility Quantification	99
4.7.3	Topology Quantification	100
4.8	Summary	100

5	Qualification of Quantitative Spatial Information	103
5.1	The Qualification Algorithm	103
5.2	Qualification of Topological Relations	104
5.3	Qualification of Visibility Relations	110
5.4	Qualification of Cardinal Direction Relations	114
5.4.1	Precise Reference Object	116
5.4.2	Imprecise Reference Object	117
5.4.3	Discussion	122
5.5	Computational Complexity of Qualification	125
5.5.1	Topology Qualification	125
5.5.2	Visibility Qualification	126
5.5.3	Cardinal Direction Qualification	127
5.6	Summary	128
6	A Hybrid Spatial Reasoning System	129
6.1	A Hybrid Qualitative-Quantitative Reasoning System	129
6.1.1	Architecture	130
6.2	Multi-calculus Constraint Network	131
6.3	The Geometric Reasoning Component	133
6.3.1	Computational Complexity of Geometric Reasoning	134
6.4	The Qualification Component	134
6.4.1	Computational Complexity of the Qualification Component	135
6.5	The Qualitative Reasoning Component	135
6.5.1	Generalization of Ternary Relations' Composition	139
6.6	The Hybrid Spatial Reasoning Algorithm	140
6.6.1	A Hybrid Spatial Reasoning Example	141
6.6.2	Heuristics for Reducing the Computation Time	143
6.6.3	Combination of Qualitative Calculi	145
6.7	Thematic Based Reduction	146
6.8	Summary	148
7	Experimental Evaluation	149
7.1	Prototype Geographic Information Integration System	149
7.1.1	Spatial Data Input, Visualization, and Querying	149
7.1.2	Storage Layer	150
7.1.3	Geographic Information Integration Layer	151
7.2	Evaluation of the Geographic Information Integration Layer	152
7.2.1	The Quantification Component	154
7.2.2	The Qualification Component	162
7.2.3	The Hybrid Spatial Reasoning System	167

7.3	Summary	174
8	Summary and Outlook	175
8.1	Summary of the Results	175
8.1.1	The Quantification Operation	176
8.1.2	The Qualification Operation	177
8.1.3	Reasoning with Mixed Representations of Spatial Knowledge . .	178
8.2	Outlook	178
A	Infinite-Region Objects: Representation and Algorithms	181
A.1	Infinite-Region Objects	181
A.2	Algorithms for Infinite-Region Objects	183
A.2.1	Infinite-Region Object Cropping	183
A.2.2	Infinite-Region Objects Intersection	184
A.2.3	Infinite-Region Objects Union and Difference	186
A.2.4	MBR, Convex Hull, and Tangents of Infinite-Region Objects . .	187
B	Testbed – Complementary Information	189
	References	191

List of Figures

1.1	Best routes for the ambulances to get to the hospital.	2
1.2	Hazardous areas reported by rescuers and distress calls.	3
2.1	Regions in \mathbb{R}^2	13
2.2	Geometric representation of a single multi-region object.	14
2.3	Half-plane representation.	14
2.4	Minimum Bounding Rectangle, Convex Hull, and Buffer of the object O	15
2.5	Intersection, union, and difference of the objects O_1 and O_2	17
2.6	Common tangents between convex objects.	18
2.7	Cardinal directions example.	26
2.8	Constraint networks.	27
2.9	Base relations of the Region Connection Calculus.	29
2.10	Frame of reference of the Cardinal Direction Calculus.	32
2.11	Projective ternary calculi.	37
2.12	Frame of reference of the Visibility Calculus.	37
3.1	<i>PEOPLES resilience framework</i> and the information gap after an extreme event.	47
3.2	Haiti earthquake.	50
3.3	Extreme event: Lack of information in the response phase.	51
3.4	Inference of a quantitative approximation for the Hazardous Area (HA).	55
3.5	Architecture of the geographic information integration system.	56
3.6	Quantification of the qualitative relation $NE(O_{HA}^*, O_{AI})$	57
3.7	Drawbacks in the definition of areas of interest.	59
3.8	Egg-yolk object O^*	61
3.9	Precision and accuracy in the representation of a spatial region.	61
3.10	Multi-region imprecise objects.	62
3.11	Qualification of relations between imprecise objects.	66
3.12	Reasoning with qualitative and geometric information.	67

4.1	A system to translate geographic information.	70
4.2	Quantification of CDC relations.	71
4.3	Quantification A_N^+ if the reference object is precise or imprecise.	72
4.4	Infinite acceptance areas and the corresponding infinite-region objects.	74
4.5	$A_R^-(O_2)$	77
4.6	IR definition of the CDC quantification over a precise reference object.	78
4.7	Line segment rays and quantifications over a line segment.	79
4.8	Quantification of cardinal direction relations – Case $A_R^+(O_2^+)$	82
4.9	Quantification of cardinal direction relations – Case $A_R^+(O_2^*)$	84
4.10	$A_V^-(O_2, O_3)$	86
4.11	Quantification of the visibility relations using infinite-region objects.	87
4.12	$A_{\mathcal{R}_{Vis}^{ST}}^+(O_2, O_3^+)$	90
4.13	$A_{\mathcal{R}_{Vis}^{ST}}^+(O_2, O_3^*)$	90
4.14	$A_{\mathcal{R}_{Vis}^{ST}}^+(O_2^+, O_3)$	91
4.15	$A_{\mathcal{R}_{Vis}^{ST}}^+(O_2^+, O_3^+)$	91
4.16	$A_{\mathcal{R}_{Vis}^{ST}}^+(O_2^+, O_3^*)$	92
4.17	$A_{\mathcal{R}_{Vis}^{ST}}^+(O_2^*, O_3)$	92
4.18	$A_{\mathcal{R}_{Vis}^{ST}}^+(O_2^*, O_3^+)$	93
4.19	$A_{\mathcal{R}_{Vis}^{ST}}^+(O_2^*, O_3^*)$	93
4.20	Quantification of RCC-8 relations.	96
5.1	Consistency checking of $R(O_1, O_2^o)$	106
5.2	Qualification of visibility relations between imprecise objects.	112
5.3	$R_{Vis}(O_i^*, O_4, O_5^*)$	113
5.4	Qualification of CDC relations between imprecise objects.	115
5.5	Cardinal direction crisp relations between imprecise objects.	119
5.6	Objects configuration.	125
6.1	Architecture of the hybrid qualitative-quantitative reasoning system.	130
6.2	Multi-calculus constraint network.	132
6.3	MN_C derived from the multi-calculus constraint network in Fig. 6.2.	132
6.4	Algebraic closure algorithm.	137
6.5	Actual object configuration and known objects.	142
6.6	First approximations of O_4^* , O_5^* , O_6^* , and O_9^*	143
6.7	Final approximations of O_4^* and O_9^*	144
6.8	Hybrid spatial reasoning system with the thematic reduction component.	147
7.1	QGIS Phyton plugin.	150
7.2	Storage layer – <i>Logical schema</i>	151

7.3	Geographic information integration layer – <i>Class diagram</i>	152
7.4	OpenStreetMap dataset – Bremen.	154
7.5	Quantification testbed.	155
7.6	Quantification of RCC relations.	157
7.7	CDC quantification: precise reference object.	158
7.8	CDC quantification: imprecise reference object with empty yolk.	158
7.9	CDC quantification: imprecise reference object with non-empty yolk.	158
7.10	Quantification of Visibility relations: precise reference objects.	160
7.11	Quantification of Visibility relations: imprecise reference objects.	160
7.12	Qualification testbed.	163
7.13	Qualification of RCC-8 relations.	164
7.14	Qualification of CDC relations.	165
7.15	Qualification of Visibility relations.	166
7.16	Hybrid spatial reasoning testbed extracted from OSM Bremen.	168
7.17	Hybrid reasoning system – Heuristics.	169
7.18	Hybrid reasoning system – Computation time.	170
7.19	Hybrid reasoning system – Reasoning iterations.	173
8.1	Overlaying of quantifications over existing maps.	177
A.1	Complex infinite-region objects.	182
A.2	Cropping of an infinite-region object.	183
A.3	Intersection of two infinite-region objects.	185
A.4	MBR and convex hull of infinite-region objects.	187
B.1	Qualification testbed.	190
B.2	Hybrid spatial reasoning testbed.	190

List of Tables

2.1	Inverse table for the RCC-8 calculus.	30
2.2	Composition table for the RCC-8 calculus.	30
2.3	The composition of CDC single-tile relations.	34
2.4	Allen's Interval Algebra.	35
2.5	Correlation among CDC rectangular relations and RA relations.	35
2.6	The inverse of CDC single-tile relations.	36
2.7	Permutation and rotation of Visibility relations.	38
2.8	Composition of two single-tile visibility relations	38
3.1	Haiti Earthquake: raster data availability.	49
3.2	Haiti Earthquake: vector data availability.	49
4.1	Quantification of Visibility relations with imprecise reference objects.	89
4.2	Quantification of RCC-8 relations with imprecise reference objects.	97
5.1	Qualification of RCC-8 relations – $R_{RCC}(O_1^*, O_2^*)$	106
5.2	Qualification of RCC-8 relations	111
5.3	Side(R^{ST}) operator.	118
5.4	Qualification of CDC relations between spatial-region objects.	123
7.1	Output relations – Variable number of unknown entities.	171
7.2	Output relations – Variable number of input relations.	171
7.3	Quantification reduction with variable number of unknown entities.	172
7.4	Quantification reduction with variable number of input relations.	172

Chapter 1

Introduction

1.1 Motivation

The development of computational approaches for the integration of spatial information is crucial for supporting rescue operations after extreme events. When a natural disaster, such as an earthquake, a flood, or a tornado, hits a city, several operations start in order to rescue injured people and, in general, give first aid to all people affected by the event. For instance, fire brigades rescue trapped people from collapsed buildings and clear the main connection roads obstructed by rubble, while red cross aims at transporting injured people to the hospitals and setting up first gathering points for delivering relief goods to the population. These operations rely on spatial knowledge about the region hit by the disaster, such as maps reporting the infrastructure's functionality after the event. Nevertheless, information that describes the environment before the event is not reliable anymore for grounding any decision since the event can cause several changes in the region: for example, after an earthquake, buildings can collapse and roads can be obstructed by rubble. Hence, *first responders* (such as fire brigades and ambulance staff) communicate with each other in order to report about the problems they encounter while accomplishing a particular task and to inform the others about critical situations. Also, *distress calls* received from people affected by the event may be used to know about critical situations that follow the natural event.

For instance, imagine that the city depicted in Fig. 1.1 has been struck by an earthquake and that the following messages are received by different emergency lines:

FIRE BRIGADES. A big fire broke out north west of the *Airport*. As far as I see it extends to the *River*.

RED CROSS. The road north of the *Stadium* is totally obstructed by rubble: it is not possible to drive there anymore.

DISTRESS CALL TO POLICE. I live in the *University College*. There is a big fire in the direction of the *Airport* but it did not clear the college yet.

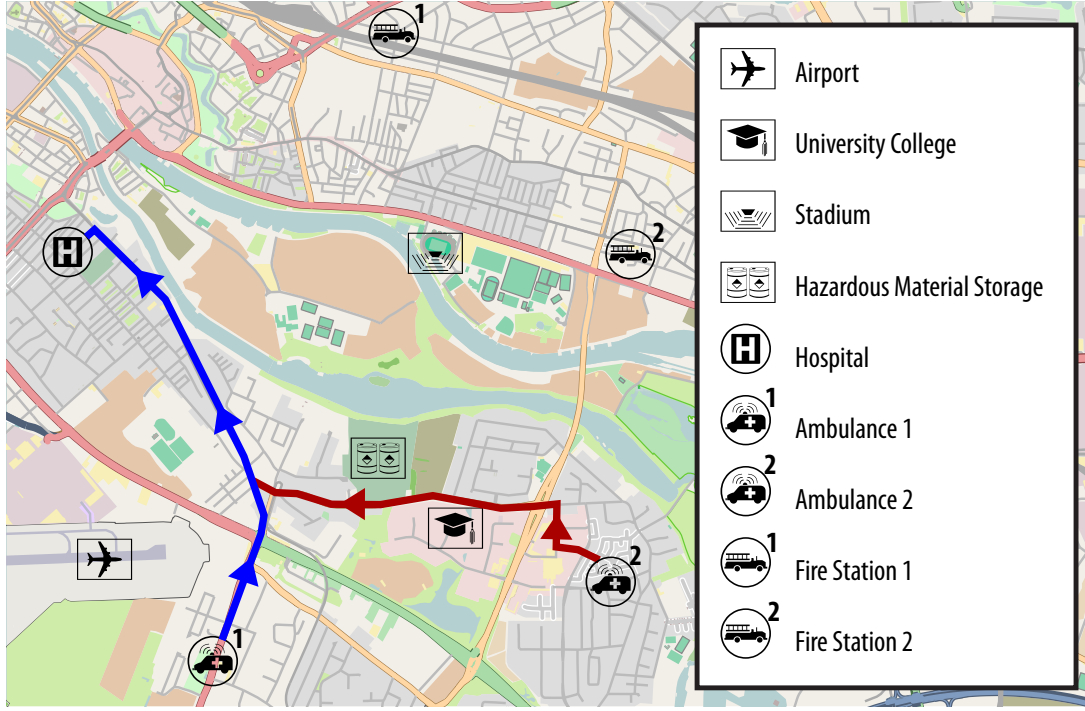


Figure 1.1: Best routes for the ambulances to get to the hospital.

Ambulances need to take wounded people to the hospital as fast as possible. In order to reach their destination, they follow the route instructions displayed on a navigation system, especially if they are not familiar with the environment¹. In the case the above communications are broadcasted to the first responders, and hence to the ambulance drivers, they would be able to realize that the routes cross particularly dangerous areas and subsequently they would search for alternative routes. For instance, the driver of *Ambulance 1* may notice that his route could be blocked by the fire since the route suggested by the system goes across the area north west of the airport, and a fire has been reported in that area by the fire brigades. To avoid additional risks, he could search beforehand for an alternative way to get to the hospital. However, it is not guaranteed that all rescuers receive the communications: for example the driver of *Ambulance 2* may not receive the reports about the fire and he would then follow the route instructions suggested by his system. Since that route crosses the area of the fire, it would be blocked by the flames and the driver would be forced to go back in order to

¹This situation arises when people from different areas and regions get to the zone hit by the catastrophe in order to help in the rescue operations.

avoid the fire area; only afterwards he can search an alternative route to the hospital. As main consequence, the driver needs more time to take the wounded to the hospital and this could tragically result in the loss of lives.

Furthermore, even if the communications do not mention it directly, the *Hazardous Material Storage* site could go up in flames. In such a case, due to the kind of material stored, the extreme event may have even more catastrophic consequences. Hence, fire brigades with proper equipment should be sent to shield the site from the fire; however, this choice depends on whether or not the receiver of the reports notices the danger on time. Similar problems are faced by all other people involved in the disaster rescue operations. To overcome these problems, the communications concerning damages and risks, such as the fire as well as the obstructed road near the stadium, should be interpreted, reported on the map and then shared with all actors involved in the rescue phases.

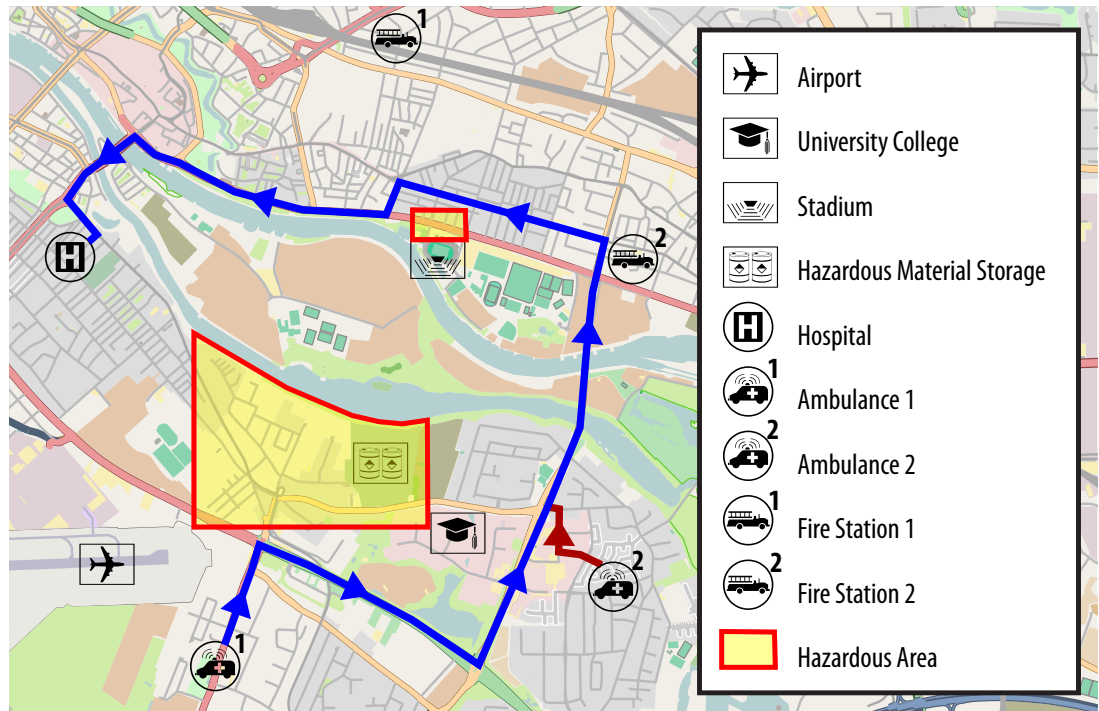


Figure 1.2: Hazardous areas reported by rescuers and distress calls.

At first instance, the interpretation of the communication can be performed by human operators. Humans are able to almost instantaneously interpret spatial information contained in verbal reports as the ones described above. However, considering the amount of information exchanged after an extreme event, it is impossible for only one person, or even for a group of persons, to analyze and interpret the big flow of communication coming from all rescuers or people affected by the disaster. Therefore, in

order to integrate the information and spread it to the first responders, a system needs to be developed that automatically: (i) collects all the communications, (ii) interprets them, (iii) integrates the information with previous knowledge, (iv) draws the results on a map and, finally, (v) shares the new information with other people involved in the rescue operations.

An example of the information that such a system yields is depicted in Fig. 1.2. The descriptions of both the fire and the obstructed road near the stadium are interpreted and depicted on the map as *Hazardous Areas*. These interpretations can be automatically spread to first responders. Hence, the drivers of the ambulances would directly see that the routes suggested by the navigation systems cross the *Hazardous Areas*: in the case they evaluate that passing through these areas presents too many risks, they could ask the system to search for an alternative route to get to the hospital. However, they could decide to ignore the alert in case they believe the risk is not so high, for instance if the route crosses the dangerous area only in a short stretch. Equivalently, the system can be set to automatically suggest an alternative path if risky conditions are met by the shortest path. Furthermore, considering that the *Hazardous Material Storage* site is exactly in the area in which the fire is burning, an automatic alert can be sent to the fire stations in order to dispatch the closest fire brigades to shield that area from the fire and avoid additional risks. In the example of Fig. 1.2, *Fire Station 2* is alerted since it is the closest fire station to the site.

The integration of spatial information aims at combining information collected from different sources and having differing conceptual representations, and at providing an unified view of the information¹. In order to perform the integration, the system has to take into account the heterogeneous characteristics of spatial information as described in a map and as reported in human communications. The integration process calls for meeting the challenges posed by the different forms of spatial information. In particular, challenges that arise by considering the imprecise² traits of spatial descriptions are faced in this work.

1.2 Problem Characterization

1.2.1 Geographic Information Systems

Geographic Information Systems (GIS) are used for storing and managing geographic data collected by national or local administrations, enterprises, and companies. The characteristics of the information depend on the specific business: information on roads and other infrastructure is gathered to draw maps or develop car navigation systems;

¹Information integration is also called *information fusion* in the literature.

²The distinction among concepts such as *precision*, *accuracy*, and *resolution* of spatial descriptions will be discussed in Section 2.3.

land use information is collected for cadastral or ecological applications; hazard maps are created to provide the population of a certain region with information on possible risks of disasters.

GIS are not designed to only store information, but they also implement algorithms to retrieve particular information that is necessary to solve a specific task. A local administration that wants to build a new road would be interested in knowing who are the owners of the parcels of land eventually crossed by the new road. A car navigation system employs such algorithms to identify the *best route* to get to a chosen destination (e.g., the hospital).

In general, high precision is required in the data stored into a GIS; the actual level of precision depends on the purpose of the information. For example, if a road is not precisely measured and is reported within a range of 50 meters around its real position, a car navigation system could not be able to locate its actual position or could produce misleading results while computing the best route. For the same reason, with a low level of precision in the data, the administration could get a wrong list of land owners.

Geographic information changes are often slow processes and the changes affect only a small part of the information stored in a GIS. For instance, the construction of a new road or building requires a long time to be completed, and it affects only the neighbor of the new entity. In this case, the information in the GIS can be updated by making accurate measurements of the new entity and entering them into the system. Additional data is collected using the same approach in order to update information related to neighboring objects that are affected by the change.

1.2.2 Geographic Information After Extreme Events

The scenario drastically changes when an unexpected natural event, like an earthquake or a tornado, modifies the typical *static* environment within a few seconds. Information changes are not slow processes anymore and can concern a big share of the whole information stored in a GIS. Nevertheless, most of the rescuing decisions following the extreme event rely on the spatial knowledge of the environment: in order to take better decisions, the information should be up-to-date. If a crucial decision is grounded on outdated information, several drawbacks can arise: for instance, as described in Section 1.1, the driver of *Ambulance 2*, that did not receive the communications that report the fire, needs more time than necessary to get to the hospital, since the route suggested by his navigation system is obstructed by the fire.

Hence, GIS information needs to be updated as soon as possible in order to provide all actors involved in the rescuing with reliable knowledge about the environment. Even though several methods have been developed to collect geographic data, such as the GPS technology, they require more than one day before the data is available, as shown after the disaster that hit the city of Port-Au-Prince (Haiti) in January 2010. For this

reason, information available in the first hours after the event, and so in the phase in which reliable information can make the difference between a successful or unsuccessful rescue, was not up-to-date and hence it was not reliable.

However, nontraditional sources of geographic information can be taken into consideration as an alternative to the common collecting methods¹. In particular, distress call and communications among rescue squads provide descriptions about changes that affected the environment. This information is usually only interpreted by the receiver and it is eventually forwarded to other rescuers: the information is hence only partially exploited. In order to provide the rescuers with up-to-date knowledge, such information should be automatically interpreted and shared with the first responders, for instance by means of maps.

The interpretation does not yield a precise description of the real extent of the entity reported in the communications (e.g., the fire) due to the lack of specificity in some of the spatial descriptions, rather an approximation of it can be computed. However, the approximate description is adequate for reducing the hazard in the decisions that are taken after an extreme event. For instance, even if the area of the fire is only approximate and is not precisely described (some zones within that area are not actually affected by the fire), the fire brigades can be automatically alerted to shield the *Hazardous Material Storage Site* from the fire. The approximate information is used to reduce the hazard in any decision, hence improving the response to the extreme event.

1.2.3 Information Integration After Extreme Events

After an extreme event, first responders ground their decisions on geographic knowledge commonly stored and managed by Geographic Information Systems. These decisions can be supported by the use of automatic instruments to retrieve specific information necessary to solve a particular task (e.g., best route computation). As discussed above, geographic information collected before the disaster is not reliable anymore. However, as shown in Section 1.1, while communicating with each other, first responders report and describe the changes in the environment caused by the extreme event. This information must be exploited to update and to enrich the rescuers' knowledge of the environment.

In order to provide rescuers, and in general all people involved in the first aid phases, with maps of the region hit by the disaster that also report the damages that it caused (like in Fig. 1.2), methods to integrate geographic information described by humans with the information stored into GIS must be developed. The characteristics of and the differences among the various information sources must be analyzed to perform the integration.

¹These approaches will be detailed in Section 3.1.

Entities managed by Geographic Information Systems are represented employing *quantitative* information: the hospital, for instance, has a unique and precise location that can be represented through its coordinates within a well-defined *geographic reference system* (e.g., longitude and latitude coordinates). On the other hand, humans do not describe geographic information as it is done in GIS: saying that the obstructed road is *north of* the stadium is different from giving the road's latitude and longitude coordinates. *Qualitative* information, that abstracts from the quantitative detail, is employed by humans to describe particular spatial relationships (e.g., *north of*) holding between specific entities (like the *stadium* and the *road*). Such information can be easily interpreted by other people, even if it does not precisely express the position of the referenced entity.

In addition, it is possible to perform *reasoning* operations over the information in order to *infer* knowledge that is not explicitly represented. *Computational geometry* algorithms have been developed to analyze and perform inference operations over quantitative information, whilst methods to reason with purely qualitative information have been developed in the research field of *Qualitative Spatial Representation and Reasoning* (QSR). These reasoning operations strengthen the capabilities of the different representation approaches: if somebody communicates that there are people to be rescued *in* the *Hazardous Material Storage* site, it is not necessary to also explicitly state that the people are *in* the area of the fire since such information can be inferred by properly connecting the available data.

The challenge of this work is to develop a computational method for the integration of qualitative information with quantitative information through the development of a reasoning system that combines inference techniques developed separately for the different kinds of information. As the separate reasoning approaches empower the specific representation methods for which they have been designed, a combined reasoning approach allows to extract implicit information from heterogeneous spatial information: quantitative and qualitative. In turn, the information inferred by the combined approach is either qualitative or quantitative.

1.3 Thesis and Contribution

The thesis of this work is that:

The development of an inference strategy to reason with heterogeneous quantitative and qualitative representations of spatial knowledge improves the integration process of geographic information collected from heterogeneous sources such as sensors and humans.

In particular, the combined inference strategy:

- is able to deal with entities only partially or imprecisely described;
- performs inference operations from information given either or both quantitatively and/or qualitatively.

The main contributions of this dissertation are:

- This work provides translation methodologies to convert quantitative information into qualitative information, and vice-versa. An extension of the representation methods used in GIS is proposed and adopted in order to cope with quantitative information generated by the interpretation of qualitative information. Likewise, approaches to deal with qualitative information are extended to cope with entities that are only approximately described.
- A geographic information integration approach is discussed that employs and extends inference techniques developed in the field of Qualitative Spatial Representation and Reasoning and algorithms developed for managing quantitative information in GIS. It is shown that the proposed method is able to enforce the spatial knowledge base, both in qualitative and in quantitative terms.
- The proposed strategies for spatial information translation and integration are empirically evaluated in order to analyze the system's outcomes.

1.4 Outline

The remainder of this text is structured as follows: Diverse approaches to represent spatial knowledge and manipulate quantitative and qualitative information are introduced in Chapter 2 along with basic notations. The chapter also describes existing works for representing imprecise spatial information.

In Chapter 3 the information availability after an extreme event is analyzed by considering as an example the earthquake that struck the city of Port-au-Prince (Haiti) in January, 2010. A system for the integration of qualitative and quantitative spatial information is proposed as a solution for providing up-to-date spatial knowledge after extreme events. Hence, the challenges in spatial information integration are identified, which provide the motivation for the work presented in the next chapters.

Chapter 4 and Chapter 5 are concerned with problem of transforming spatial information from a representation approach to the other. The former discusses the problem of translation from qualitative information to quantitative one, while the latter focuses

on extracting qualitative spatial knowledge from the quantitative description of spatial entities. Imprecision in spatial information is considered in the transformation approaches, and different aspects of the space are considered.

Chapter 6 is dedicated to the development of the reasoning system that deals with mixed quantitative-qualitative representations of spatial knowledge. It enforces reasoning techniques developed separately to deal with either qualitative or quantitative information.

The experimental evaluation of the proposed integration approach is described in Chapter 7. The system's outcomes are evaluated by considering real input datasets.

Finally, Chapter 8 summarizes the outcomes of this work and provides an overview on open questions and further research following this work.

Chapter 2

Geographic Information Manipulation

In this chapter, key concepts with regard to geographic knowledge representation and reasoning approaches will be briefly summarized, and the notations used in the remainder of this text are introduced. Section 2.1 is an introduction to the *quantitative* representation approaches commonly used in existing *Geographic Information Systems*, and some algorithms to manipulate quantitative information are discussed. Furthermore, different methods to collect geospatial information are analyzed. Yet, in Section 2.2 the background concepts in the area of *qualitative spatial reasoning* are summarized, with a particular focus on those concepts necessary for the scope of this thesis. Finally, in Section 2.3 different approaches developed to take into account *imprecision* in spatial information are analyzed.

This chapter provides only the basic concepts with regard to spatial representation and reasoning. Challenges in the integration of different kinds of spatial information for emergency management will be then discussed in Chapter 3.

2.1 Quantitative Representations and Computational Geometry

Geographic Information Systems (GIS) have been successfully employed to solve geographic problems in several domains, such as route planning, cadastral applications, and emergency management (see Longley *et al.* (2005) for a complete introduction to Geographic Information Systems and Science). There does not exist in the literature a shared definition of what a GIS is, and several attempts to define what a GIS should and should not include have been done since the 1980s (e.g., Cowen, 1988). The acronymous GIS usually refers to a system of hardware and software elements that allows for

collecting, storing, managing, and transforming geographic data; GIS provides functions for the interaction with end users also (e.g., Nyerges & Jankowski (2010); Rigaux *et al.* (2002)). In particular, the focus of this work will be on the so-called *spatial databases* (Rigaux *et al.*, 2002), that are a specific kind of databases designed to store spatial information (Goodchild, 1989). In this text, the term GIS is used to refer both to the geographic analysis tools and to the spatial databases beneath.

GIS make use of *quantitative* approaches to represent and manipulate spatial entities. An entity is represented in a GIS by its coordinates within a specific *geographic reference system*. The analysis of the different reference systems is not in the scope of this work, an overview can be found in Iliffe & Lott (2008).

In the remainder of this text, the notation *spatial entities* will be used to denote items or phenomena existing in the reality that occupy *spatial regions*; in contrast, *spatial objects* stand for the representation of spatial regions¹.

2.1.1 Quantitative Representations

Models for representing spatial entities can be classified into two different categories: *continuous fields* and *discrete objects*² (Longley *et al.*, 2005). The former is concerned with entities whose properties to represent vary continuously in the space (such as soil composition) and are commonly represented in GIS using *raster* approaches³. In contrast, the latter is concerned with entities whose properties to represent do not vary within the entity's boundary (such as buildings); these kinds of entities are usually represented using *vector* methods. A different mode for the representation of discrete objects is the *half-plane representation*, even though this approach is not implemented in most of the existing GIS tools. Since the entities considered in this work belong to the category of discrete objects, in the remainder of this text only the vector representation and the half-plane representation will be considered.

2.1.1.1 Vector Representation

In compliance with the OpenGIS specifications (Herring, 2001), vector representations employ parameterized primitive objects such as points, lines, and polygons to represent the extension of spatial entities in different dimensions. In this work, exclusively 2D geometric objects embedded in the Euclidean space \mathbb{R}^2 are employed. Most of the primitives and operations used are supported by standard geographic information systems.

¹Whenever the spatial nature of the entity is clear from the context, the terms *entity*, *region*, and *object* will be used in place of respectively spatial entity, spatial region, and spatial object.

²Continuous fields and discrete objects are also called respectively *field-based models* and *entity-based models* in the literature (e.g., Rigaux *et al.*, 2002; Worboys & Duckham, 2004).

³The raster representation is based on a partition of the space into regular cells, e.g., squares.

In the following the notations used for the primitive geometric objects occurring in the text are introduced.

Points in the plane are denoted with a single small letter and are identified by their Cartesian coordinates, e.g., $p = (x_p, y_p)$. *Lines* are represented by two points p_1 and p_2 and are denoted with a small Greek letter, e.g., $\lambda = (p_1, p_2)$. *Line segments* consist of all points that lie between p_1 and p_2 on the line $\lambda = (p_1, p_2)$ (including p_1 and p_2 themselves) and are written as $\bar{\lambda} = [p_1, p_2]$, while $\vec{\lambda} = [p_1, p_2)$ is used for *rays*, which are oriented line segments with start point p_1 and end point p_2 but extending beyond p_2 into infinity. *Polylines* are finite sequences of connected line segments and denoted by capital Greek letters. They are specified either as lists of line segments or as point lists, e.g., $\Lambda = \langle \bar{\lambda}_1, \bar{\lambda}_2, \dots, \bar{\lambda}_n \rangle$ or $\Lambda = \langle p_1, p_2, \dots, p_{n+1} \rangle$.

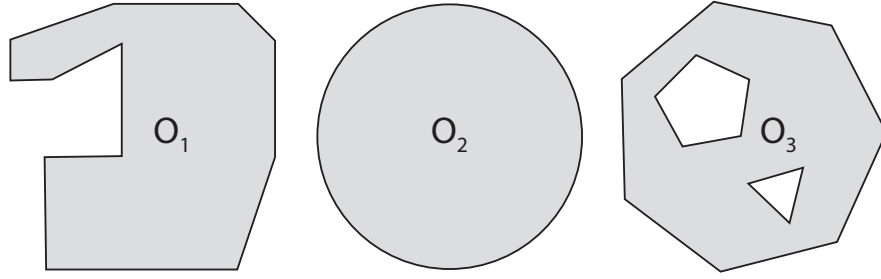


Figure 2.1: Regions in \mathbb{R}^2 .

Regions in \mathbb{R}^2 can be grouped into two different categories: the first class contains all regions that are closed, connected, and have closed boundaries, while the second class contains also the regions that are disconnected or that contain holes. Adopting the formalism of Skiadopoulos & Koubarakis (2004), the former class will be denoted as *REG* and the latter as *REG**. The concave object O_1 and the convex object O_2 depicted in Fig. 2.1 represent regions that belong to both *REG* and *REG**, while the disconnected object composed by the union of O_1 and O_2 , and the object with holes O_3 represent regions belonging only to *REG**.

Simple-region objects (also called *polygons*) are used to represent regions in *REG* adopting the conceptualization of a region as a point-set. The representing polygons are defined as closed polylines and denoted by a capital letter, e.g., $O = \langle p_1, p_2, \dots, p_n, p_1 \rangle$. In order to distinguish a polygon O from its boundary, a function $\Delta(O)$ is used to denote the polygon boundary, that is hence a polyline. To be able to deal with general spatial regions that may have several disconnected components and holes (regions in *REG**), *multi-region objects* (also called *multi-polygons*) are employed, as defined in the OGC standard (Herring, 2001). A complex polygonal object with holes is specified by a list of simple polygons of which the first polygon represents the outer boundary of the region, while the other polygons describe the non-overlapping holes, e.g., $C = \langle P, Q_1, \dots, Q_n \rangle$. A multi-region object composed of n disconnected parts is then written as a list of

polygonal objects (potentially with holes), e.g., $M = \langle C_1, C_2, \dots, C_n \rangle$. An example of a multi-region object is depicted in Fig. 2.2.

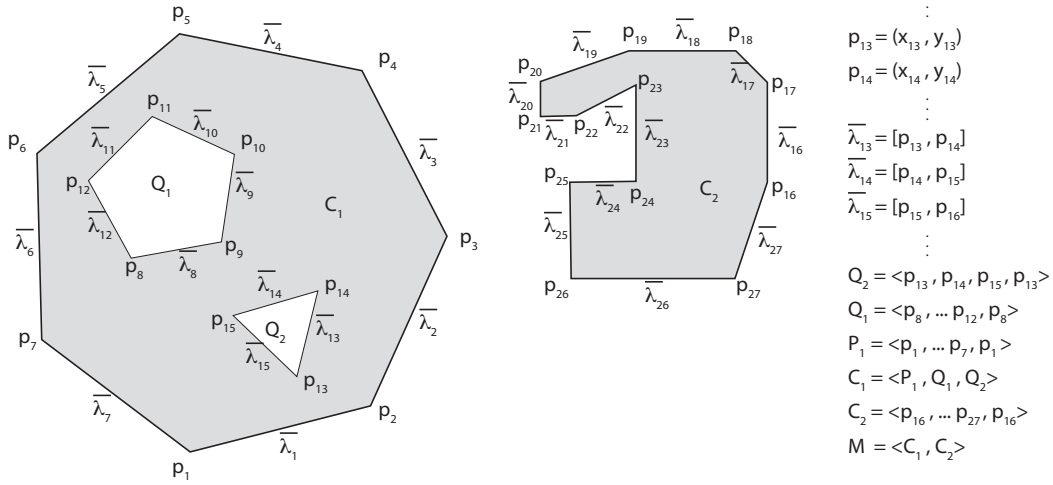


Figure 2.2: Geometric representation of a single multi-region object.

The symbols O_1, \dots, O_n will be employed in this text to refer to general spatial objects that can be either simple-region or multi-region objects.

2.1.1.2 Half-plane Representation

The half-plane representation makes use of a single geometrical primitive: the *half-plane* (Rigaux & Scholl, 1995; Rigaux *et al.*, 2002). Formally, a half-plane is defined as the set of points $p = (x_p, y_p)$ in \mathbb{R}^2 that satisfy the inequality $ax_p + by_p + c \leq 0$. For instance, let $\lambda = (p_1, p_2)$ be the *directed* line from p_1 to p_2 —mathematically represented by the equation $ax_\lambda + by_\lambda + c = 0$ —the half-plane that it represents, denoted as $HP = [p_1, p_2]$, is the shaded region in Fig. 2.3(a).

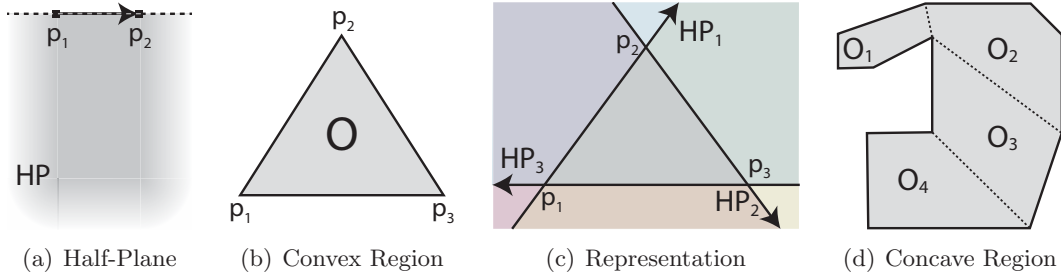


Figure 2.3: Half-plane representation.

A convex region can be represented as the intersection of a finite number of half-planes. As an example, the region depicted in Fig. 2.3(b) is represented using half-planes as $O = \langle HP_1, HP_2, HP_3 \rangle = \langle [p_1, p_2], [p_2, p_3], [p_3, p_1] \rangle$ (Fig. 2.3(c)). In contrast, a non-convex region can not be represented directly as the intersection of a finite number of half-planes. Rather, the region is firstly decomposed into a set of convex adjacent components and then represented as the union of the different components. For instance, the region in Fig. 2.3(d) can be decomposed into the convex regions represented by O_1, O_2, O_3 and O_4 , then the four components are represented using the half-planes and, finally, $O = O_1 \cup O_2 \cup O_3 \cup O_4$. The decomposition into convex components is not unique, and it is performed in $\mathcal{O}(n)$ time in the worst case, where n is the number of points that define the spatial object O .

2.1.2 Computational Geometry Algorithms

Computational geometry techniques are employed to generate new information and perform analyses by manipulating geometric objects that convey quantitative information about the spatial extension of spatial entities, for instance by way of computing intersections, unions, and convex hulls (see de Berg *et al.* (2008); O'Rourke (1998) for an overview on computational geometry algorithms).

The main geometric operations used in this text are the union, intersection, and set difference of spatial objects as well as operations that compute the minimum bounding rectangle, the convex hull, and the buffer of a given object, and the common tangents between two objects.

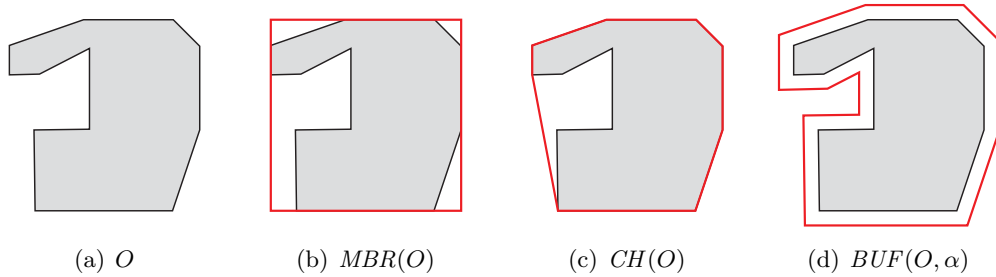


Figure 2.4: Minimum Bounding Rectangle, Convex Hull, and Buffer of the object O .

2.1.2.1 Minimum Bounding Rectangle

Given a spatial object O defined by n points, the *Minimum Bounding Rectangle*—denoted as $MBR(O)$ —is the axis-aligned minimal rectangle that contains O (Fig. 2.4(b)). The rectangle is computed by identifying the minimum and maximum coordinate values of the points that define the spatial object. Hence, it can be trivially proven that

the computation of the *MBR* requires $\mathcal{O}(n)$ time. Furthermore, $\overline{X}(O)$ is used to denote the x-axis maximum value of O , $\underline{X}(O)$ the x-axis minimum value, $\overline{Y}(O)$ the y-axis maximum value, and $\underline{Y}(O)$ denotes the y-axis minimum value; they are formally defined as:

$$\begin{aligned}\overline{X}(O) &= \max(x_i), \forall (x_i, y_i) \in O & \underline{X}(O) &= \min(x_i), \forall (x_i, y_i) \in O \\ \overline{Y}(O) &= \max(y_i), \forall (x_i, y_i) \in O & \underline{Y}(O) &= \min(y_i), \forall (x_i, y_i) \in O\end{aligned}$$

2.1.2.2 Convex Hull

The *Convex Hull* of a set of n points S , denoted as $CH(S)$, is the minimal convex set containing S . Hence, given a region represented by a spatial object O , the convex hull $CH(O)$ is the convex hull of the set of points that defines O . One of the most used algorithms that computes the convex hull in $\mathcal{O}(n)$ time has been presented in Melkman (1987). Fig. 2.4(c) shows the convex hull of a concave object O . The convex hull of a region in *REG* or *REG** is always a simple region in *REG*.

2.1.2.3 Buffering

Given a spatial object O —convex or concave and eventually containing holes—defined by n points, and a real value α , the *buffering* operation¹—denoted as $BUF(O, \alpha)$ —computes the object whose boundaries are at a distance equals to α from the boundary of O , resulting thus in an inflation/deflation of the original object. Fig. 2.4(d) shows the buffering of O with a positive value α . The buffering operation can be based either on the *Minkowski sum* (Wein, 2007), resulting in a spatial object containing arcs, or on the *straight skeleton*² that result in an object defined only by straight lines. In this work, only the second approach is considered. An algorithm to compute the buffering based on the straight skeleton has been presented in Felkel & Obdrzalek (1998) and it runs in $\mathcal{O}(n^2)$ time in the worst case.

2.1.2.4 Intersection, Union, and Difference

Traditional intersection, union, and set difference symbols \cap , \cup , \setminus are used to denote the respective homonymous operations between spatial objects as described in Herring (2001). When applying the operation to two objects of a particular type, it is assumed that the resulting point set is always returned as an object of the same type, i.e., given two multi-region objects O_1, O_2 , the operation $O_1 \cap O_2$ yields a new multi-region object. As an example, Fig. 2.5 depicts the objects resulting from the application of

¹This operation is also called *polygon offset* in the literature.

²The approach based on the straight skeleton is also called *Miter offset*.

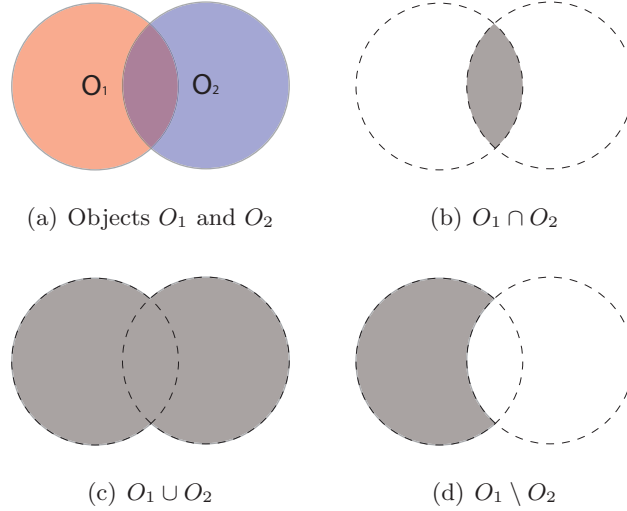


Figure 2.5: Intersection, union, and difference of the objects O_1 and O_2 .

the different operators on two simple objects O_1 and O_2 . Let O_1 be defined by n points and O_2 be defined by m points, intersection, union, and difference algorithms (Margalit & Knott, 1989; Shan-xin & Rui-lian, 2010) perform in $\mathcal{O}(nm)^1$ time in the worst case.

2.1.2.5 Common Tangents

As shown in Clementini & Billen (2006), given two convex spatial objects O_1 (n points) and O_2 (m points), it is possible to uniquely identify two pairs of common tangents between them: the *internal* common tangents, that intersect inside $CH(O_1 \cup O_2)$, and the *external* common tangents that intersect outside $CH(O_1 \cup O_2)$. The common tangents between two convex objects are depicted in Fig. 2.6. The external tangents (dotted lines) are differentiated as *External Tangent Right (ER)*—that is the tangent for which the half-plane defined by its points of tangent on O_1 and O_2 contains both O_1 and O_2 —and *External Tangent Left (EL)*—the half-plane defined by its points of tangent on O_1 and O_2 does not contain the objects. Similarly, the two internal tangents (solid lines) are distinguished as *Internal Tangent Right (IR)*—that is the tangent for which the half-plane defined by its points of tangent on O_1 and O_2 contains only O_2 —and *Internal Tangent Left (IL)*—the half-plane defined by its points of tangent on O_1 and O_2 contains only O_1 . An algorithm that computes the tangents between convex

¹The computational complexity in (Margalit & Knott, 1989; Shan-xin & Rui-lian, 2010) is $\mathcal{O}(n + m + k)$, with k being the number of intersections of the edges of the two objects. It can be trivially proved that in the worst case k is equal to nm . Hence, the algorithm performs in $\mathcal{O}(nm)$ time in the worst case.

spatial objects in $\mathcal{O}(\log(n + m))$ time in the worst case is described in Kirkpatrick & Snoeyink (1995).

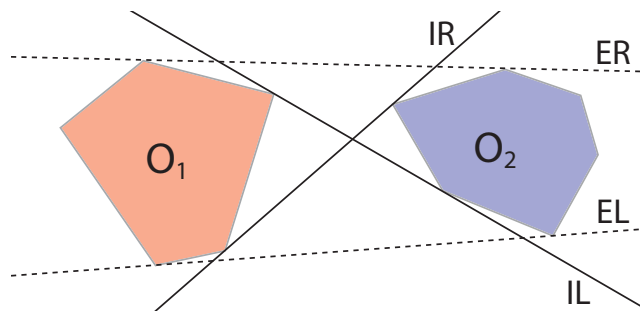


Figure 2.6: Common tangents between convex objects.

2.1.3 Data Collection Methods

In this section, traditional methods commonly used to collect geographic data are introduced. These methods reflect the state of the art with regard to the source of data for GIS. Subsequently, some innovative collection methods are presented.

2.1.3.1 Traditional Collection Methods

According to Longley *et al.* (2005), the methods to collect geographic data for GIS can be classified in two main categories:

PRIMARY METHODS. Data is collected either directly from field measurements or from aerial and satellite pictures. Data is captured specifically for use in GIS by directly measuring the physical properties of spatial entities.

SECONDARY METHODS. Data is collected from existing sources, such as topographic map, photographs, and other hard-copy documents. Hence, data is captured from earlier collections or it is obtained from other systems.

In particular, methods belonging to the *primary* class are:

SURVEY MEASUREMENTS. They are based on ground measurements that determine angles and distances of an entity with respect to other known points on the earth surface. Electro-optical devices replaced the traditional theodolites, that were used by surveyors to measure angles, and tapes and chains, to measure distances. Ground survey is an expensive and time-consuming activity, but it is still the way to obtain the most accurate point locations.

GPS MEASUREMENTS. Global Positioning System (GPS) consists of 24 satellites, each orbiting the Earth every 12 hours on distinct orbits, and transmitting radio pulses at very precise time intervals. A receiver determines its position by making precise calculations from the received signals, the known position of the satellites, and the signal velocity. Positioning in three dimensions requires at least four satellites above the horizon. The accuracy of the measurement depends on the number of visible satellites and their position. Differential GPS (DGPS) improves the accuracy combining GPS signals from satellites with correction signals received via radio or telephone from base stations. However, the accuracy of the measurement degrades in cities with tall buildings, or under trees, and GPS signals are totally lost under bridges or in indoor environments.

SATELLITE REMOTE SENSING IMAGES AND AERIAL PHOTOGRAPHS. Remote sensors derive information about physical properties of spatial entities without physical contact; they measure the amount of electromagnetic radiation reflected, emitted, or scattered from physical entities. Usually, those sensors are fixed on Earth-orbiting satellites or aircrafts. Although both of them are grounded on the same technology, satellite images are more suitable for large scale data collection, with a correspondent loss in spatial resolution; aerial photographs are used for small scale projects in which a better resolution is required (e.g., LIDAR). Different pictures can be interpolated in order to create 3D coordinates, contours, and Digital Elevation Models (DEM) of entities. Aerial and satellite images are often geo-referenced using points gathered from ground surveys.

In comparison, the *secondary methods* class contains:

DATA CAPTURE USING SCANNERS. Hard-copy media, such as building plans, CAD drawings, or photographs, are transformed into digital images through a process of scanning. Digitization improves the access to the data through the usage of integrated database storage and geographic indexing. Furthermore, useful information can be extracted from the digitized media, such as building footprints or road networks.

MANUAL AND HEADS-UP VECTORIZATION. Vector objects are digitized from maps or other geographic data sources. This operation can be performed manually, using specific devices—called digitizers—to capture specific points on a map, or can be partially automated by instructing the digitizer to collect points automatically based on some spatial constraints. Heads-up vectorization is performed using a computer to collect points instead of a digitizer. Software to perform automated vectorization can speed up the process of point collection; however, vectorization software introduces errors that need to be manually edited.

PHOTOGRAMMETRY. It has been defined by Longley *et al.* (2005) as the “*science and technology of making measurements from pictures, aerial photographs, and images*”. Photographs are geo-referenced using control points defined by ground survey or collected using GPS. Usually, 3-D properties of entities are extracted with these methods, such as digital elevation models.

2.1.3.2 Volunteered Geographic Information

The aforementioned methods for geographic data collection require specific devices to collect data (usually affordable only by enterprises due to their high costs) and expertise knowledge in the geographic field. However, as a result of the diffusion of cheap technologies—such as GPS devices—people are now allowed to collect and to share geographic data without the necessity of expensive devices and expertise knowledge in the geography field. This led to the development of the so-called *Volunteered Geographic Information* (VGI) applications (Goodchild *et al.*, 2007), such as OpenStreetMap¹ and WikiMapia², in which data is collected and shared through the internet by volunteers instead of by specialized companies. Moreover, studies in the *crowdsourcing* field (Howe, 2008) have shown how information obtained from many observers can be, in some cases, more reliable than information collected from only one source.

Even though existing VGI systems already represent strong platforms for volunteers to provide and query spatial information, the systems still demand for structured and quite precise data. However, sensors do not always provide precise data (e.g., noisy measures), and frequently sensor networks are not able to gather data that describes all spatial aspects of the measured entities (e.g., satellite imagery does not provide information about the height of the buildings). Besides, the way humans communicate about space rarely carries structured data, such as numerical and geometric descriptions of spatial entities. Hence, techniques to represent and deal with unstructured and imprecise spatial descriptions have to be developed in order to improve the capabilities of VGI.

2.2 Qualitative Spatial Representation and Reasoning

Quantitative descriptions of spatial knowledge (e.g., maps) only represent some aspects of the geographic space (Berendt *et al.*, 1998). Accordingly, geographic data collection is shifting from traditional methods, commonly based on quantitative descriptions, towards new user-centered methods, that hence should reflect the way humans describe the space. Considering for instance the example introduced in Section 1.1, the actors

¹OpenStreetMap: <http://www.openstreetmap.org/>

²WikiMapia: <http://wikimapia.org/>

involved in the rescue phases after a disaster describe the space using *qualitative relations* between spatial entities (e.g., north of) instead of giving precise measurements and locations for the entities they are reporting. These qualitative descriptions abstract from the numerical description adopted into GIS.

In order to develop strategies to exploit qualitative descriptions of spatial knowledge, the Qualitative Spatial Representation and Reasoning (QSR) field has developed as a subfield of Artificial Intelligence. QSR outcomes have been applied during the years to solve several problems related to GIS, robot navigation, computer vision, and natural language interpretation (Cohn & Hazarika, 2001). For some overviews on the QSR field outcomes, see Cohn (1997); Cohn & Hazarika (2001); Cohn & Renz (2008).

QSR aims at capturing human-level concepts of space by using finite sets of *relations* to model particular spatial aspects such as topology, orientation, distance. Qualitative relations abstract from the quantitative details but preserve the information relevant for spatial decision processes and reasoning (Freksa *et al.*, 2000). QSR techniques allow to reason with spatial knowledge even in those cases in which exact quantitative description is not available (Cohn, 1997).

Several qualitative spatial *calculi*—each composed by a set of relations and some operations over them—have been developed to model specific aspects of the space. The calculi can be firstly classified considering the different aspects of the space they focus on, like *topology* (Cohn *et al.*, 1997; Egenhofer & Franzosa, 1991; Papadias *et al.*, 1995; Randell *et al.*, 1992), *relative orientation* (Billen & Clementini, 2004, 2006; Freksa, 1992; Guesgen, 1989; Ligozat, 1993; Moratz *et al.*, 2000, 2005; Renz & Mitra, 2004; Schlieder, 1995), *cardinal directions* (Balbiani *et al.*, 1998; Frank, 1991; Goyal & Egenhofer, in press), *distance* (Hernández *et al.*, 1995), *visibility* (Fogliaroni *et al.*, 2009; Tarquini *et al.*, 2007; Tassoni *et al.*, 2011), *shape* (Billen *et al.*, 2002; Clementini & Di Felice, 1997b; Galton & Meathrel, 1999), and *size* (Gerevini & Renz, 2002; Raiman, 1991; Zimmermann, 1995).

Besides the modeled aspects, a fundamental difference to consider among the calculi is the spatial ontology they are grounded on: *points in 2D space* (Frank, 1991; Freksa, 1992; Ligozat, 1993; Moratz *et al.*, 2005; Renz & Mitra, 2004), *lines in 2D space* (Moratz *et al.*, 2000; Schlieder, 1995), *2D rectangles* (Balbiani *et al.*, 1998; Guesgen, 1989), *regions in 2D space* (Clementini & Billen, 2006; Fogliaroni *et al.*, 2009; Goyal & Egenhofer, in press; Randell *et al.*, 1992; Tarquini *et al.*, 2007), *3D regions* (Billen & Clementini, 2006; Tassoni *et al.*, 2011), and *regions in n-dimensional space* (Egenhofer, 1991).

Directional calculi are traditionally further classified by considering their *frame of reference* (FoR). According to the classification provided by Levinson (1996), the FoR can be:

INTRINSIC. The properties of the objects (such as oriented lines) are used to define the direction relations (Moratz *et al.*, 2000; Schlieder, 1995).

RELATIVE. The direction relations are defined through a third object as anchoring point for the frame of reference (Billen & Clementini, 2004; Freksa, 1992; Ligozat, 1993; Moratz *et al.*, 2005).

ABSOLUTE. The direction is defined by an external point of reference, such as the earth north pole (Frank, 1991; Goyal & Egenhofer, in press; Renz & Mitra, 2004).

In the remainder of this section, the characteristics of qualitative calculi will be formally defined and algorithms developed to reason with qualitative descriptions of spatial knowledge will be introduced. Furthermore, three qualitative calculi that model three different aspects of the space—namely topology, cardinal direction, and visibility properties—will be described. Since any qualitative calculus models only a single aspect of the space, approaches to merge qualitative calculi when different aspects are considered will be discussed. The section will be concluded with a description of existing methods to translate spatial information from quantitative to qualitative descriptions (and vice-versa), and the state of the art in qualitative modeling and GIS will be analyzed.

2.2.1 Qualitative Spatial Calculi

A *qualitative spatial calculus* is defined by a set of relations between spatial objects¹ and a set of operations defined on these relations. In this section, only the concepts necessary in the scope of this work will be introduced, while more detailed explanations and discussions can be found in Ligozat & Renz (2004); Renz & Ligozat (2005); Renz & Nebel (2007). A formal definition of qualitative relations, operations over qualitative relations and qualitative spatial calculi will be given in Section 2.2.1.1 and Section 2.2.1.2; a toy qualitative spatial calculus will be discussed in Section 2.2.1.3 to provide examples for the formally defined concepts.

2.2.1.1 Qualitative Relations

Let $\mathcal{D} = (O_1, \dots, O_m)$ be a potentially infinite domain, a *qualitative relation* is:²

Definition 1 (*n*-ary Qualitative Relation).

An *n*-ary qualitative relation R over a domain \mathcal{D} is a subset of the *n*-ary Cartesian product of the domain: $R \subseteq \mathcal{D} \times \dots \times \mathcal{D} = \mathcal{D}^n$.

¹The mapping between spatial regions and spatial objects is a *homomorphism* (see for instance Robinson & Frank, 1985). Thus, a qualitative relation between spatial objects corresponds to the qualitative relation between the corresponding spatial regions in the reality.

²In this text, the notation $R(O_1, O_2, \dots, O_n)$ and the expressions O_1, \dots, O_n *satisfy* or *verify* the relation R , are used equivalently to denote $(O_1, \dots, O_n) \in R$.

When the domain \mathcal{D} is finite, it is possible to *extensionally* define any relation by listing all n -tuples that belong to the relation. The spatial case is more complex since the domain contains all points, lines and regions in the 2D space, and hence it is infinite; in such a case, the relations need to be *intensionally* defined by listing the properties that the n -tuples have to satisfy in order to belong to a certain relation.

For the purpose of this work, it is important to distinguish between the reference objects, that define the frame of reference, and the primary object of the relation.

Definition 2 (Primary and reference objects of a qualitative relation).

Given $O_1, \dots, O_n \in \mathcal{D}$, with $(O_1, \dots, O_n) \in R$, O_1 is named the primary object of the n -tuple O_1, \dots, O_n in the relation R and O_2, \dots, O_n are the reference objects of the n -tuple O_1, \dots, O_n in R .

Let \mathcal{B} be a set of n -ary qualitative relations, the concept of *JEPD set of relations* is introduced to denote the case in which every n -tuple of objects that belongs to the domain \mathcal{D} satisfies one and only one relation in the set \mathcal{B} .

Definition 3 (JEPD set of n -ary qualitative relations).

A set of n -ary qualitative relations \mathcal{B} over a domain \mathcal{D} is called Jointly Exhaustive and Pairwise Disjoint (*JEPD*) if the following conditions are satisfied:

$$(i) \bigcup_{R \in \mathcal{B}} R = \mathcal{D}^n \quad (ii) \forall R_1, R_2 \in \mathcal{B}, R_1 \cap R_2 = \emptyset \text{ with } R_1 \neq R_2$$

The relations belonging to a JEPD set of relations \mathcal{B} are named *base* relations. The complete set of relations $\mathcal{R}_{\mathcal{B}}$, named *general* relations, can be generated from the set of JEPD base relations \mathcal{B} . The resulting set of all possible unions of base relations is called *the set of general relations* of the qualitative calculus. In this text, the notation $\{R_1, R_2, \dots, R_m\}$ is used to denote the general relation that is the union of the base relations R_1, R_2, \dots, R_m . Furthermore, \emptyset denotes the empty relation, while U refers to the general relation formed by all base relations in \mathcal{B} , and is called *universal relation*. A relation in the form $\{R_1, \dots, R_m\}$ —called *disjunctive* relations—, with $m \geq 2$, represents incomplete qualitative knowledge because it is not exactly known which is the base relation holding between the primary and the reference objects.

2.2.1.2 Operations on Qualitative Relations

Classical set operations of *complement* ($^{\circ}$), *union* (\cup), and *intersection* (\cap) can be applied to the set of general relations $\mathcal{R}_{\mathcal{B}}$.

Definition 4 (Complement, union, and intersection).

Given $R, R_1, R_2 \in \mathcal{R}_B$, complement, union and intersection operations are defined as:

$$R^\circ = \{x \mid x \in U \wedge x \notin R\} \quad (\text{Complement})$$

$$R_1 \cup R_2 = \{x \mid x \in R_1 \vee x \in R_2\} \quad (\text{Union})$$

$$R_1 \cap R_2 = \{x \mid x \in R_1 \wedge x \in R_2\} \quad (\text{Intersection})$$

Other operations can be defined over a set of general n -ary relations \mathcal{R}_B : unary *permutation* operations—that given an n -ary relation yield the relations satisfied by particular permutations of any n -tuple in the relation—and binary *composition* operations—that given two n -ary relations satisfied by two overlapping sets of objects, infer which relations are satisfied by any n -ary composition of the sets of objects. In the binary case, only one permutation operation exists. Given a general binary relation between the objects O_1 and O_2 , the unary *inverse* operation (\sim) yields the relation between O_2 and O_1 .

Definition 5 (Inverse of a binary relation).

Given a binary relation $R \in \mathcal{R}_B$ over a domain \mathcal{D} , the unary inverse (\sim) operation is defined as: $R^\sim = \{(O_1, O_2) \in \mathcal{D}^2 \mid (O_2, O_1) \in R\}$

In the ternary case, six different permutation operations can be defined, as discussed in Freksa & Zimmermann (1993).

Definition 6 (Permutations of a ternary relation).

Given a ternary relation $R \in \mathcal{R}_B$ over a domain \mathcal{D} , the six unary permutation operations are defined as follows:

$$ID(R) = \{(O_1, O_2, O_3) \in \mathcal{D}^3 \mid (O_1, O_2, O_3) \in R\} \quad (\text{Identity})$$

$$INV(R) = \{(O_2, O_1, O_3) \in \mathcal{D}^3 \mid (O_1, O_2, O_3) \in R\} \quad (\text{Inverse})$$

$$SC(R) = \{(O_1, O_3, O_2) \in \mathcal{D}^3 \mid (O_1, O_2, O_3) \in R\} \quad (\text{Shortcut})$$

$$SCI(R) = \{(O_3, O_1, O_2) \in \mathcal{D}^3 \mid (O_1, O_3, O_2) \in R\} \quad (\text{Shortcut inverse})$$

$$HM(R) = \{(O_2, O_3, O_1) \in \mathcal{D}^3 \mid (O_1, O_3, O_2) \in R\} \quad (\text{Homing})$$

$$HMI(R) = \{(O_3, O_2, O_1) \in \mathcal{D}^3 \mid (O_1, O_3, O_2) \in R\} \quad (\text{Homing inverse})$$

Besides, the binary *composition* operation (\circ) is an operation that yields the relation between O_1 and O_3 , given the relations holding respectively between O_1 and O_2 and between O_2 and O_3 .

Definition 7 (Composition of binary relations).

Given two binary relations $R_1, R_2 \in \mathcal{R}_B$, their binary composition is defined as:

$$R_1 \circ R_2 = \{(O_1, O_3) \in \mathcal{D}^2 \mid \exists O_2 \in \mathcal{D} : (O_1, O_2) \in R_1 \wedge (O_2, O_3) \in R_2\}$$

In this text, the composition of binary relations will be also denoted as $R(O_1, O_3) = R_1(O_1, O_2) \circ R_2(O_2, O_3)$. The results of the composition operation are not always relations in \mathcal{R}_B ; in order to perform *symbolic operations*, meaning operations that only operate over the symbols that represent the relations, and do not consider the elements of the relations, the concept of composition has been relaxed into the *weak composition* operation.

Definition 8 (Binary weak composition).

Given two binary relations $R_1, R_2 \in \mathcal{R}_B$, their binary weak composition is defined as:

$$R_1 \diamond R_2 = \{R \in \mathcal{B} \mid R \cap (R_1 \circ R_2) \neq \emptyset\}$$

Composition and weak composition operation of ternary relations can be similarly defined. However, as shown in Freksa (1992), there exist different ways to combine the relations in the ternary case. Nevertheless, all compositions can be expressed by a permutation of composing two permutations of relations Scivos & Nebel (2001). Since there exist different possibilities to compose ternary relations, in the remainder of this text the operations will be explicitly defined when required. The same notation as the composition of binary relations will be used.

Permutation and composition operations can be defined also for n -ary qualitative relations (Condotta *et al.*, 2006); however, such operations are not necessary in the scope of this paper, since the focus will be only on binary and ternary relations. The operations are generally defined by means of tables, that get the name of *reasoning tables*. Permutation are expressed by tables of dimension $1 \times n$ —being n the number of base relations—, while composition operations are tables with dimension $n \times n$. The result of the composition of two general relations is given by the union of the composition of the single base relations that compose them, for instance $\{R_1, R_2\} \circ \{R_3, R_4\} = (R_1 \circ R_2) \cup (R_1 \circ R_3) \cup (R_2 \circ R_3) \cup (R_2 \circ R_4)$.

Finally, a *qualitative spatial calculus* is defined by a set \mathcal{B} of base relations over a domain \mathcal{D} of spatial objects, and a set of operations (complement, union, intersection, permutation, and composition) on those relations, that enable elementary reasoning operations and form the basis for more complex reasoning procedures.

2.2.1.3 Qualitative Spatial Calculus: a Toy Example

An exemplary domain $\mathcal{D} = \{O_1, O_2, O_3\}$, composed by three spatial objects is considered. The configuration of the three spatial objects is depicted in Fig. 2.7(a). Cardinal direction relations between those objects can be extensionally defined as

$N = \{(O_1, O_2)\}$, $S = \{(O_2, O_1)\}$, $W = \{(O_2, O_3)\}$, $E = \{(O_3, O_2)\}$, $NW = \{(O_1, O_3)\}$, $SE = \{(O_3, O_1)\}$ and $B = \{(O_1, O_1), (O_2, O_2), (O_3, O_3)\}$. The same relations can be intensionally defined by imposing some constraints on the coordinates of the objects¹. The relations can be equivalently expressed as $N(O_1, O_2)$, $S(O_2, O_1)$, etc. Furthermore, if the objects of Fig. 2.7(b) are considered in place of the domain \mathcal{D} , the cardinal direction relations do not undergo any change, even if the objects have different shapes with respect to the ones in Fig. 2.7(a).

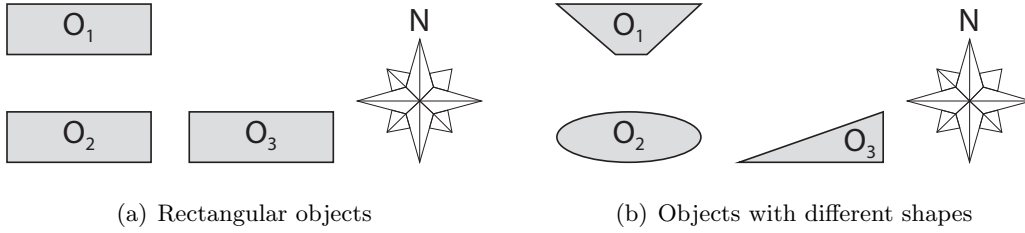


Figure 2.7: Cardinal directions example.

It can be trivially demonstrated that the set of relations $\mathcal{B}_{CD} = \{N, S, W, E, NW, SE, B\}$ is JEPD. Some examples of general relations belonging to \mathcal{R}_{CD} , that can be generated from \mathcal{B}_{CD} , are: $\{N, E\} = \{(O_1, O_2), (O_3, O_2)\}$, $\{N, S\} = \{(O_1, O_2), (O_2, O_1)\}$. Furthermore, $U_{CD} = \{N, S, W, E, NW, SE, B\} = \{(O_1, O_1), (O_1, O_2), (O_1, O_3), (O_2, O_1), (O_2, O_2), (O_2, O_3), (O_3, O_1), (O_3, O_2), (O_3, O_3)\} = \mathcal{D}^2$.

The complement, union, and intersection operations, for some of the defined relations, yield the following results:

$$B^\circ = \{(O_1, O_2), (O_1, O_3), (O_2, O_1), (O_2, O_3), (O_3, O_1), (O_3, O_2)\}$$

$$B \cup N = \{(O_1, O_1), (O_1, O_2), (O_2, O_2), (O_3, O_3)\}$$

$$B \cap N = \{\} = \emptyset$$

Moreover, the inverse operation yields: $N^\sim = \{(O_2, O_1)\} = S$, $S^\sim = \{(O_1, O_2)\} = N$, $W^\sim = \{(O_3, O_2)\} = E$, $E^\sim = \{(O_2, O_3)\} = W$, $NW^\sim = \{(O_3, O_1)\} = SE$, $SE^\sim = \{(O_1, O_3)\} = NW$ and $B^\sim = \{(O_1, O_1), (O_2, O_2), (O_3, O_3)\} = B$. Furthermore, the composition operation returns the following results: $N \circ W = \{(O_1, O_3)\}$, $N \circ B = \{(O_1, O_2)\}$, $N \circ S = \{(O_1, O_1)\}$, etc. In the first two cases, the result of the composition is a relation in \mathcal{R}_{CD} — $N \circ W = NW$, $N \circ B = N$ —while in the third case the composition's result is not a relation. In contrast, adopting the weak composition, the results are: $N \diamond W = NW$, $N \diamond B = N$, and $N \diamond S = B$.

¹Formal and complete definitions of the cardinal direction relations will be introduced in Section 2.2.3.2.

Finally, the qualitative binary spatial calculus CD is defined as: $\mathcal{D} = (O_1, O_2, O_3)$, $\mathcal{B}_{CD} = \{N, S, W, E, NW, SE, B\}$, and the operations $^\circ, \cup, \cap$ defined as in Definition 4, \sim as in Definition 5, and \circ as in Definition 7.

2.2.2 Representation of Spatial Configurations

As shown in Section 2.2.1.3, the *spatial configuration* of a set of objects can be qualitatively represented using the relations—defined into one ore more calculi—holding between them. Even though the qualitative representations do not uniquely identify a specific set of spatial objects¹, they can rather be interpreted as constraints restricting the possible geometries the related objects can adopt. Spatial configurations can be represented qualitatively by means of the so-called *constraint networks*.

2.2.2.1 Constraint Networks

Given a qualitative spatial calculus, a qualitative spatial representation is a set of constraints expressed in a quantifier-free constraint language based on the set \mathcal{R}_B of general relations. It can be seen as a *constraint network* $N = (V, \mathcal{D}, R)$ with variables $V = \{O_1, O_2, \dots, O_n\}$ over the domain \mathcal{D} whose valuations are constrained by relations given in the constraint matrix R ($R_{O_i O_j}$ gives the qualitative relation—constraint—between O_i and O_j). The constraint network derived from the spatial configuration of Section 2.2.1.3 is depicted in Fig. 2.8(a).

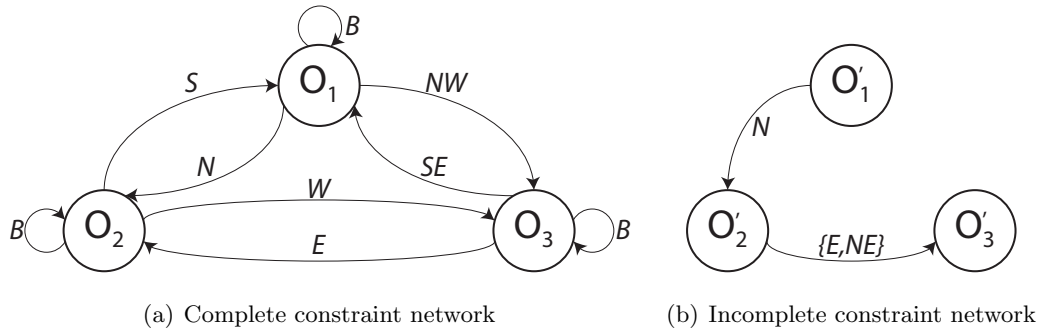


Figure 2.8: Constraint networks.

The network of Fig. 2.8(a) is *complete*, in the sense that any node of the network is connected to any other node (and they are even self-connected) and any label has only one base relation of the exemplary calculus CD . This is verified because all objects in the domain have a well-defined geometric description, and hence it is possible to

¹The relations defined in Section 2.2.1.3, for instance, are valid for the objects in both Fig. 2.7(a) and Fig. 2.7(b).

compute the actual relation between any pair of them. However, this is not always the case: indeed, as introduced in Section 2.1.3.2, spatial data can be gathered through human descriptions—that carry imprecise and incomplete knowledge—, or through sensors that can be either noisy or do not measure all aspects of a certain spatial entity. However, human descriptions and sensors’ information can be interpreted in qualitative terms without the necessity to know the exact quantitative descriptions of the spatial entities. Some objects can hence be under-specified or uncertain. By taking unions of relations as constraints, one can express uncertainty. For instance, considering a spatial domain consisting of three objects O_1^* , O_2^* , and O_3^* , for which the geometries are not known (hence the domain of the objects is defined by all regions in \mathbb{R}^2) but for which it is known that O_1^* is somewhere north of O_2^* and that O_2^* is somewhere east or north-east of O_3^* , the configuration of those objects can be represented through a constraint network adopting the relations defined for the exemplary calculus CD , as Fig. 2.8(b) shows. If no edge connects two nodes, no information is known to relate the objects and it can be modeled with the universal relation U_{CD} . The two given examples show constraint networks for binary calculi, however all concepts described in this section can be directly transferred to ternary calculi, and, more in general, to n -ary calculi.

2.2.2.2 Reasoning Problems

A spatial constraint network has a *solution* if one can assign objects from the domain to the variables such that all constraints are satisfied. One important reasoning problem is to decide whether a spatial constraint network is *consistent* which means it has a solution. This problem is an instance of the *constraint satisfaction problem*. In the case of typical qualitative spatial calculi, the domain is infinite and, hence, techniques for solving finite constraint satisfaction problems are not directly applicable.

The consistency can be checked based on a procedure called *algebraic closure* or *path consistency algorithm*, firstly introduced in Montanari (1974) for binary calculi and then refined by Mackworth (1977), that uses the composition and converse operations to enforce consistency of all triples of variables V_i, V_j, V_k by performing $R_{ik} = R_{ik} \cap (R_{ij} \circ R_{jk})$ until a fixpoint is reached or a resulting relation becomes empty which indicates inconsistency. Dylla & Moratz (2004) discussed an algorithm, based on Mackworth (1977), to check consistency for ternary calculi. The algorithm employs the composition of ternary relations defined as $R(O_1, O_2, O_4) = R_1(O_1, O_2, O_3) \circ R_2(O_2, O_3, O_4)$.

The algebraic closure algorithm for binary calculi defined by Mackworth (1977) performs in $\mathcal{O}(n^3)$ time in the worst case, being n the number of variables in the constraint network. The worst case computational complexity becomes $\mathcal{O}(n^4)$ time if ternary calculi are considered.

2.2.3 Topology, Cardinal Directions, and Visibility Calculi

In the remainder of this work, three qualitative spatial calculi dealing with different aspects of space will be considered. The three calculi deal with regions in \mathbb{R}^2 which makes them particularly useful for GIS applications. The three calculi have been chosen to model *topology*, *cardinal direction*, and *visibility* spatial aspects of the space.¹ In particular, topological and cardinal direction relations are binary relations, while visibility relations are ternary ones.

2.2.3.1 Region Connection Calculus

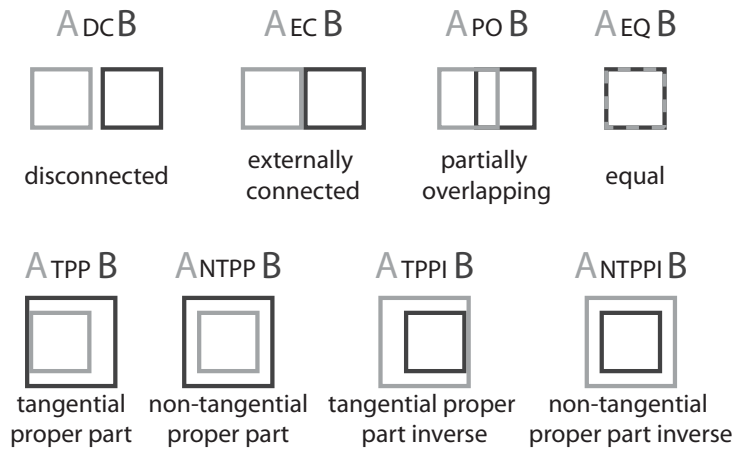


Figure 2.9: Base relations of the Region Connection Calculus RCC-8.

Different binary calculi suitable to represent topological relations between extended regions in 2D have been proposed, among which the most popular were firstly introduced by Egenhofer & Franzosa (1991); Randell *et al.* (1992).

Randell *et al.* (1992) introduce the *RCC-8*² theory, that is based on the concept of *connectedness* between spatial entities. Several ways in which connectedness can be defined are discussed by Cohn & Varzi (1999). Eight base relations are logically derived from this concept in order to represent topological relation: *DC* (disconnected), *EC* (externally connected), *PO* (partially overlapping), *EQ* (equal), *TPP* (tangential proper part), *NTPP* (non-tangential proper part), *TPPI* (tangential proper part inverse) and *NTPPI* (non-tangential proper part inverse). The set of base JEPD topological relation is $\mathcal{B}_{RCC} = \{DC, EC, PO, EQ, TPP, NTPP, TPPI, NTPPI\}$. As a result of its definition, the RCC-8 calculus is not only suited for spatial entities,

¹The choice of those three calculi follows the analysis of the Emergency Management scenario that will be discussed in Chapter 3.

²For the sake of readability, in the remainder of this text RCC-8 calculus will be also denoted as *RCC*.

Table 2.1: Inverse table for the RCC-8 calculus.

R	DC	EC	PO	EQ	TPP	NTPP	TPPI	NTPPI
R[~]	DC	EC	PO	EQ	TPPI	NTPPI	TPP	NTPP

but also for all those entities (either physical or conceptual) for which the concept of connectedness can be defined. Fig. 2.9 illustrates the RCC-8 relations for 2D spatial objects. A reduced version of the RCC-8 calculus has been proposed by Cohn *et al.* (1997), which considers only five topological relation and gets the name of *RCC-5*. Differently from the RCC-8, the RCC-5 groups the *TPP* and *NTPP* relations into the *PP* (proper part) relation and *TPPI* and *NTPPI* ones into the *PPI* (proper part inverse) relation. The rotation operation for the RCC-8 calculus can be easily defined from the semantics of the relations, as shown in Table. 2.1. Yet, the composition table has been described by Cui *et al.* (1993) and it is shown in Table. 2.2. Furthermore, it has been proofed by Düntsch *et al.* (2001) that the composition of RCC-8 relations is a weak form of composition.

Table 2.2: Composition table for the RCC-8 calculus.

R₁ \ R₂	DC	EC	PO	EQ	TPP	NTPP	TPPI	NTPPI
DC	\mathcal{B}_{RCC}	DC, EC, PO, TPP, NTPP	DC, EC, PO, TPP, NTPP	DC	DC, EC, PO, TPP, NTPP	DC, EC, PO, TPP, NTPP	DC	DC
EC	DC, EC, PO, TPPI, NTPPI	DC, EC, PO, EQ, TPP, TPPI	DC, EC, PO, EQ, TPP, NTPP	EC	EC, PO, TPP, NTPP	PO, TPP, NTPP	DC, EC	DC
PO	DC, EC, PO, TPPI, NTPPI	DC, EC, PO, TPPI, NTPPI	\mathcal{B}_{RCC}	PO	PO, TPP, NTPP	PO, TPP, NTPP	DC, EC, PO, TPPI, NTPPI	DC, EC, PO, TPPI, NTPPI
EQ	DC	EC	PO	EQ	TPP	NTPP	TPPI	NTPPI
TPP	DC	DC, EC	DC, EC, PO, TPP, NTPP	TPP	TPP, NTPP	NTPP	DC, EC, PO, TPP, TPPI, NTPPI	DC, EC, PO, TPPI, NTPPI
NTPP	DC	DC	DC, EC, PO, TPP, NTPP	NTPP	NTPP	NTPP	DC, EC, PO, TPP, NTPP	\mathcal{B}_{RCC}
TPPI	DC, EC, PO, TPPI, NTPPI	EC, PO, TPPI, NTPPI	PO, TPPI, NTPPI	TPPI	PO, TPP, TPPI, NTPPI	PO, TPP, NTPP	TPPI, NTPPI	NTPPI
NTPPI	DC, EC, PO, TPPI, NTPPI	PO, TPPI, NTPPI	PO, TPPI, NTPPI	NTPPI	PO, TPPI, NTPPI	PO, EQ, TPP, NTPP, TPPI, NTPPI	NTPPI	NTPPI

In contrast, the calculus proposed by Egenhofer (1989); Egenhofer & Franzosa (1991) is based on the intersection of the interiors (O°), the boundaries (δO) and

the exteriors (O^{-1}) of two given n -dimensional objects in an n -dimensional space. The intersections yield to the Boolean *nine intersection matrix* of Equation 2.1—the model is hence called *Nine Intersection Model (9I-M)*.

$$9I-M(O_1, O_2) = \begin{bmatrix} \delta O_1 \cap \delta O_2 & \delta O_1 \cap O_2^\circ & \delta O_1 \cap O_2^{-1} \\ O_1^\circ \cap \delta O_2 & O_1^\circ \cap O_2^\circ & O_1^\circ \cap O_2^{-1} \\ O_1^{-1} \cap \delta O_2 & O_1^{-1} \cap O_2^\circ & O_1^{-1} \cap O_2^{-1} \end{bmatrix} \quad (2.1)$$

The matrix gives theoretically rise to 2^9 (512) different possible configurations; however, considering connected regions in 2D space, only eight configurations are physically realizable. Those configurations are equivalent to the relations defined by the RCC-8 calculus. The model has been further investigated by considering regions with holes (Egenhofer *et al.*, 1994). Finally, Egenhofer (1991) describes an algorithm that computes the composition of two base 9I-M topological relations.

2.2.3.2 Cardinal Direction Calculus

The concept *North of* has been defined by Fisher (2000) as: “one object lies somewhere vaguely to the north of another as opposed to south, east or west of it”. Hence, the concept can be *vaguely*¹ interpreted; for this reason, a precise semantics has to be assigned to each cardinal direction in order to avoid possible misinterpretation of the information. Different binary qualitative calculi have been proposed to model and to deal with cardinal directions (Balbiani *et al.*, 1998; Frank, 1991; Goyal & Egenhofer, in press; Guesgen, 1989).

Frank (1991) introduces a point-based model for cardinal directions. In contrast, Balbiani *et al.* (1998); Goyal & Egenhofer (in press); Guesgen (1989) are grounded on 2D spatial regions. However, in Balbiani *et al.* (1998); Guesgen (1989) the spatial objects are approximated with their Minimum Bounding Rectangle². The model of Goyal & Egenhofer (in press)—called *Cardinal Direction Calculus* (CDC)—does not introduce any approximation of the spatial objects, and can be used also to represent relations between points and lines (Goyal & Egenhofer, 2000). It is hence more convenient for representing cardinal direction relations between spatial objects and, for this reason, only the CDC will be discussed in this text.

Given two spatial regions in REG^3 represented by the spatial objects O_1 and O_2 , the CDC defines an absolute frame of reference for the cardinal direction relations as a partition of the plane into nine regions (called *acceptance areas* or *tiles*) grounded on the Minimum Bounding Rectangle of the reference object O_2 . The CDC frame of reference is

¹A definition of *vagueness* is given in Section 2.3.

²The Minimum Bounding Rectangle has been defined in Section 2.1.2.1.

³The model presented here is valid also for regions with holes, that belong to REG^* . An extension for the model that includes all regions in REG^* is discussed in Skiadopoulos & Koubarakis (2004).

shown in Fig. 2.10. Being the frame of reference based on projective properties of the objects (Frank, 1996), the CDC relations will be referred to as *projective relations* as well. The cardinal relation holding between the two objects, denoted by $R_{CDC}(O_1, O_2)$ can be represented as a Boolean matrix reporting the intersection of the primary object with the different acceptance areas (Equation 2.2).

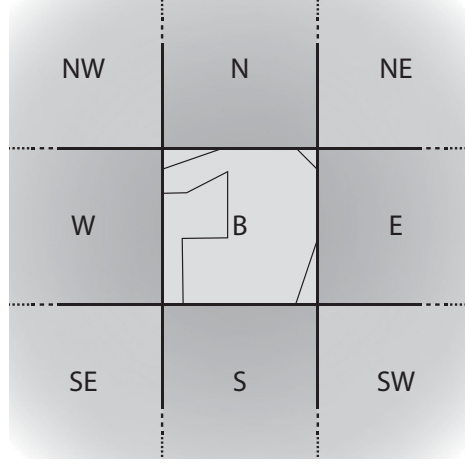


Figure 2.10: Frame of reference of the Cardinal Direction Calculus.

$$R_{CDC}(O_1, O_2) = \begin{bmatrix} O_1 \cap NW(O_2) & O_1 \cap N(O_2) & O_1 \cap NE(O_2) \\ O_1 \cap W(O_2) & O_1 \cap B(O_2) & O_1 \cap E(O_2) \\ O_1 \cap SW(O_2) & O_1 \cap S(O_2) & O_1 \cap SE(O_2) \end{bmatrix} \quad (2.2)$$

The matrix yields 2^9 (512) different possible configurations; however, only 218 possible relations are physically realizable when connected regions are taken into account (Cicerone & Di Felice, 2004). Hence, the set of JEPD cardinal direction base relations (\mathcal{B}_{CDC}) is composed by 218 different relations. Among them, the nine specific relations for which the correspondent direction matrix has only one true value—hence the primary object intersects only one acceptance area—are called *Single-Tile* (ST) relations, and they are denoted respectively with the symbols corresponding to the intersected tile: $N, NE, E, SE, S, SW, W, NW, B$. The other relations are instead called *Multi-Tile* (MT) relations, to highlight that the primary object overlaps more than one acceptance area. Multi-tile relations will be denoted as the sequence of the tiles overlapped by the primary object, with a colon as separator ($R = T_1 : \dots : T_k$, with $k \leq 9$). For instance, the relation $N:NE:NW(O_1, O_2)$ means that the object O_1 overlaps the acceptance areas—based on $MBR(O_2)$ — N, NE and NW . In the remainder of this text, R^{ST} will be used to denote a single-tile relation, while R^{MT} denotes a multi-tile relation; if it is not specified whether a relation is single or multi-tile, the generic symbol R will be used. Furthermore, the set of all CDC single-tile relations will be denoted as $\mathcal{R}_{CDC}^{ST} = \{N, NE, E, SE, S, SW, W, NW, B\}$, and $\Gamma(R^{MT})$ is a function to retrieve the tiles that compose a multi-tile relation.

The CDC base single-tile relations have been intensionally defined in Skiadopoulos & Koubarakis (2004) as:

$$B(O_1, O_2) \quad \text{iff} \quad \underline{X}(O_2) \leq \underline{X}(O_1) \wedge \overline{X}(O_1) \leq \overline{X}(O_2) \wedge \underline{Y}(O_2) \leq \underline{Y}(O_1) \wedge \overline{Y}(O_1) \leq \overline{Y}(O_2) \quad (2.3)$$

$$N(O_1, O_2) \quad \text{iff} \quad \overline{Y}(O_2) \leq \underline{Y}(O_1) \wedge \underline{X}(O_2) \leq \underline{X}(O_1) \wedge \overline{X}(O_1) \leq \overline{X}(O_2) \quad (2.4)$$

$$NE(O_1, O_2) \quad \text{iff} \quad \overline{X}(O_2) \leq \underline{X}(O_1) \wedge \overline{Y}(O_2) \leq \underline{Y}(O_1) \quad (2.5)$$

$$E(O_1, O_2) \quad \text{iff} \quad \overline{X}(O_2) \leq \underline{X}(O_1) \wedge \underline{Y}(O_2) \leq \underline{Y}(O_1) \wedge \overline{Y}(O_1) \leq \overline{Y}(O_2) \quad (2.6)$$

$$SE(O_1, O_2) \quad \text{iff} \quad \overline{X}(O_2) \leq \underline{X}(O_1) \wedge \overline{Y}(O_1) \leq \underline{Y}(O_2) \quad (2.7)$$

$$S(O_1, O_2) \quad \text{iff} \quad \overline{Y}(O_1) \leq \underline{Y}(O_2) \wedge \underline{X}(O_2) \leq \underline{X}(O_1) \wedge \overline{X}(O_1) \leq \overline{X}(O_2) \quad (2.8)$$

$$SW(O_1, O_2) \quad \text{iff} \quad \overline{X}(O_1) \leq \underline{X}(O_2) \wedge \overline{Y}(O_1) \leq \underline{Y}(O_2) \quad (2.9)$$

$$W(O_1, O_2) \quad \text{iff} \quad \overline{X}(O_1) \leq \underline{X}(O_2) \wedge \underline{Y}(O_2) \leq \underline{Y}(O_1) \wedge \overline{Y}(O_1) \leq \overline{Y}(O_2) \quad (2.10)$$

$$NW(O_1, O_2) \quad \text{iff} \quad \overline{X}(O_1) \leq \underline{X}(O_2) \wedge \overline{Y}(O_2) \leq \underline{Y}(O_1) \quad (2.11)$$

Binary composition of CDC relations

The binary composition of two base cardinal direction relations (Definition 7) can be computed by performing the algorithm proposed by Skiadopoulos & Koubarakis (2004). At first, the composition of any two single-tile relations R_1^{ST} and R_2^{ST} is presented by means of a composition table (see Tab. 2.3). The function $\delta(T_1, \dots, T_k)$ returns the disjunctive relation composed by all the single and multi-tile relations that can be obtained combining the single tiles T_1, \dots, T_k . As an example, $\delta(NE, E, SE)$ returns the disjunctive relation $\{NE, E, SE, NE:E, E:SE, NE:E:SE\}$; note that $NE:SE$ is not a realizable CDC relation for connected regions and hence is not a member of the resulting relation. In order to combine a single-tile relation with a multi-tile relation, the concept of *rectangular relation* is required: a base relation R is rectangular if there exist two rectangles a and b such that $R(a, b)$ holds. The nine single-tile relations are rectangular, and there exist 27 multi-tile rectangular relations. The composition of a single-tile relation R^{ST} with a multi-tile rectangular relation $R^{Rec} = T_{21} : \dots : T_{2k}$ is computed as:

$$R^{ST} \circ (T_{21} : \dots : T_{2k}) = \delta(R^{ST} \circ T_{21}, \dots, R^{ST} \circ T_{2k}) \quad (2.12)$$

Let $R_1 = T_{11} : \dots : T_{1j}$ and $R_2 = T_{21} : \dots : T_{2k}$ be two base relations, R_1 *includes* R_2 iff $\{T_{11}, \dots, T_{1j}\} \subseteq \{T_{21}, \dots, T_{2k}\}$. A function $Br(R)$ is introduced to define the smallest rectangular relation that includes a base relation R . Furthermore, let R^{ST} be a single-tile relation and let R^{Rec} be a rectangular relation, a function $Most(R^{ST}, R^{Rec})$ is defined that returns the rectangular relation formed by the R^{ST} -most tiles of R^{Rec} . For instance, $Most(N, N:NE:E:B)$ yields the rectangular relation $N:NE$. The defined functions are used to compute the composition between a single-tile relation R^{ST} and

Table 2.3: The composition of CDC single-tile relations.

$R_1 \backslash R_2$	N	NE	E	SE	S
N	N	NE	$\delta(NE,E)$	$\delta(NE,E,SE)$	$\delta(N,S,B)$
NE	$\delta(N,NE)$	NE	$\delta(NE,E)$	$\delta(NE,E,SE)$	$\delta(N,NE,E,SE,S,B)$
E	$\delta(N,NE)$	NE	E	SE	$\delta(SE,S)$
SE	$\delta(N,NE,E,SE,S,B)$	$\delta(NE,E,SE)$	$\delta(E,SE)$	SE	$\delta(SE,S)$
S	$\delta(N,S,B)$	$\delta(NE,E,SE)$	$\delta(E,SE)$	SE	S
SW	$\delta(N,S,SW,W,NW,B)$	U_{CDC}	$\delta(E,SE,S,SW,W,B)$	$\delta(SE,S,SW)$	$\delta(S,SW)$
W	$\delta(N,NW)$	$\delta(N,NE,NW)$	$\delta(E,W,B)$	$\delta(SE,S,SW)$	$\delta(S,SW)$
NW	$\delta(N,NW)$	$\delta(N,NE,NW)$	$\delta(N,NE,E,W,NW,B)$	U_{CDC}	$\delta(N,S,SW,W,NW,B)$
B	N	NE	E	SE	S

$R_1 \backslash R_2$	SW	W	NW	B
N	$\delta(SW,W,NW)$	$\delta(W,NW)$	NW	$\delta(N,B)$
NE	U_{CDC}	$\delta(N,NE,E,W,NW,B)$	$\delta(N,NE,NW)$	$\delta(N,NE,E,B)$
E	$\delta(SE,S,SW)$	$\delta(E,W,B)$	$\delta(N,NE,NW)$	$\delta(E,B)$
SE	$\delta(SE,S,SW)$	$\delta(E,SE,S,SW,W,B)$	U_{CDC}	$\delta(E,SE,S,B)$
S	SW	$\delta(SW,W)$	$\delta(SW,W,NW)$	$\delta(S,B)$
SW	SW	$\delta(SW,W)$	$\delta(SW,W,NW)$	$\delta(S,SW,W,B)$
W	SW	W	NW	$\delta(W,B)$
NW	$\delta(SW,W,NW)$	$\delta(W,NW)$	NW	$\delta(N,W,NW,B)$
B	SW	W	NW	B

a multi-tile relation R^{MT} (see Skiadopoulos & Koubarakis (2004) for a proof of the equation):

$$R^{ST} \circ R^{MT} = R^{ST} \circ Most(R^{ST}, Br(R^{MT})) \quad (2.13)$$

As last step, the composition between a multi-tile relation with a basic relation has to be computed. To do so, the additional concept $tile\text{-}union(R_1, \dots, R_n)$ needs to be defined as the base relation consisting of all tiles in R_1, \dots, R_n . The algorithm to compose a multi-tile relation $R^{MT} = T_1 : \dots : T_k$ with a base relation R^B is described in Algorithm 1.

Algorithm 1 CDC-Compose(R^{MT}, R^B)

```








 $R_c = \emptyset$ 
for  $T_i \in R^{MT}$  do  $S_i = T_i \circ R^B$  end for
for  $(s_1, \dots, s_k) \in S_1 \times \dots \times S_k$  do
   $R \leftarrow tile\text{-}union(s_1, \dots, s_k)$ 
  if  $R \in \mathcal{B}_{CDC}$  then  $R_c \leftarrow R_c \cup R$  end if
end for
return  $R_c$ 

```

Inverse of CDC relations

A procedure to compute the inverse of CDC base relations has been firstly described in Skiadopoulos & Koubarakis (2005). However, the proposed procedure relies on the algorithm to check the consistency of CDC constraints: given a base relation $R_1(O_1, O_2)$, the inverse relation R_1^\sim can be identified by checking the consistency between $R_1(O_1, O_2)$ and $R_2(O_2, O_1)$, for each $R_2 \in \mathcal{B}_{CDC}$. In contrast, Cicerone & Di Felice (2004) propose an algorithm that computes the inverse by solving the so-called *pairwise-consistency problem*. However, as shown in (Chen *et al.*, 2010), this method is not able to give always the correct answer. Recently, Chen *et al.* (2010); Wang & Hao (2010) introduce two similar procedures to directly compute the inverse of CDC relations exploiting the correlation between the rectangular relations of the CDC calculus with the Allen Interval Algebra (IA) (Allen, 1983), that has been developed for one-dimensional temporal intervals, and the corresponding Rectangle Algebra (RA) (Balbiani *et al.*, 1998; Guesgen, 1989), for 2D spatial regions.

Table 2.4: Allen’s Interval Algebra.

Name	before	meets	overlaps	starts	during	finishes	equal
$\mathbf{R}(\mathbf{I}_1, \mathbf{I}_2)$	$<$	m	o	s	d	f	$=$
$\mathbf{R}^{-1}(\mathbf{I}_2, \mathbf{I}_1)$	$>$	mi	oi	si	di	fi	$=$
Example							

Allen (1983) describes 13 binary base relations between temporal intervals, by analyzing the ordering among their end points (Tab. 2.4). Balbiani *et al.* (1998) uses IA relations to describe the configuration of two rectangles in the space: a RA relation is a pair of IA relations (R_x, R_y) with $R_x, R_y \in \mathcal{R}_{IA}$, with R_x being the IA relation holding between the projection of the rectangles on the x-axis, and R_y the relation on the y-axis projection. Being $R, S \in \mathcal{R}_{IA}$, the inverse of $(R \times S)$ can be computed as $(R \times S)^\sim = R^\sim \times S^\sim$. Tab. 2.5 shows the correlation among the 36 CDC rectangular relations and the RA relations.

Table 2.5: Correlation among CDC rectangular relations and RA relations.

$\begin{matrix} \text{R}_x \\ \text{R}_y \end{matrix}$	$\{<, m\}$	$\{o, fi\}$	$\{di\}$	$\{=, s, d, f\}$	$\{si, oi\}$	$\{>, mi\}$
$\{<, m\}$	SW	S:SW	SE:S:SW	S	SE:S	SE
$\{o, fi\}$	W:SW	S:SW:W:B	E:SE:S:SW:W:B	S:B	E:SE:S:B	E:SE
$\{di\}$	SW:W:NW	N:S:SW:W:NW:B	U_{CDC}	N:B:S	N:NE:E:SE:S:B	NE:E:SE
$\{=, s, d, f\}$	W	W:B	E:W:B	B	E:B	E
$\{si, oi\}$	W:NW	N:W:NW:B	N:NE:E:W:NW:B	N:B	N:NE:E:B	NE:E
$\{>, mi\}$	NW	N:NW	N:NE:NW	N	N:NE	NE

Table 2.6: The inverse of CDC single-tile relations.

R	N	NE	E	SE	S	SW	W	NW
R^\sim	S SE:S S:SW S:SE:SW	SW	W NW:W W:SW NW:W:SW	NW	N N:NE N:NW N:NE:NW	NE	E NE:E E:SE NE:E:SE	SE

The algorithm to retrieve the inverse of a CDC base relation is shown in Algorithm 2. The functions $RA(R)$ and $CDC(s, t)$ used in the algorithm return respectively the RA relation that corresponds to a rectangular CDC relation, and the CDC relation corresponding to a RA one. Furthermore, $Org(R^*)$ is a function that returns all CDC base relations for which $Br(R) = R^*$. As an example, the resulting inverse of the CDC single-tile relations are shown in Tab. 2.6; the inverse of $\{B\}$ is not reported in the table since it produces a disjunction of 198 base relations.

Algorithm 2 CDC-Inv(r)

```

Inv  $\leftarrow \emptyset$ ;  $R \leftarrow Br(r)$ ;  $(R_x, R_y) \leftarrow RA(R)$ ;  $S \times T \leftarrow (R_x, R_y)^\sim$ 
for  $s \in S$  do
  for  $t \in T$  do  $R^* \leftarrow CDC(s, t)$ ;  $Inv \leftarrow Inv \cup Org(R^*)$  end for
end for
return  $Inv$ 

```

2.2.3.3 Visibility Calculus

The ternary Visibility Calculus (VC) (Fogliaroni *et al.*, 2009; Tarquini *et al.*, 2007), defined for connected regions in REG with non-overlapping convex hulls, has been built on the same primitives—namely the external and internal tangents¹ between two spatial objects—that have been already exploited to create the projective *5-intersection model* (5IM)² (Billen & Clementini, 2004) for relative directions (Fig. 2.11(a)). Let O_1, O_2 and O_3 be spatial objects that represent regions in REG , the visibility calculus defines ternary projective relations— $R_{Vis}(O_1, O_2, O_3)$ —that describe whether and how the observer object O_3 perceives the observed object O_1 if the object O_2 is acting as an obstacle.

Tarquini *et al.* (2007) introduce a partition of the space into three acceptance areas (tiles), as depicted in Fig. 2.11(b), corresponding to the concepts of *Visible* (V), *Partially Visible* (PV), and *Occluded* (Oc). The acceptance areas are constructed based on

¹The internal and external mutual tangents between spatial object have been defined in Section 2.1.2.5.

²The 5-intersection model defines five single-tile relations corresponding to the directional concepts of *After* (Af), *Before* (Bf), *Left side* (Ls), *Right side* (Rs), and *Between* (Bw). A discussion about the 5IM is not in the scope of this text.

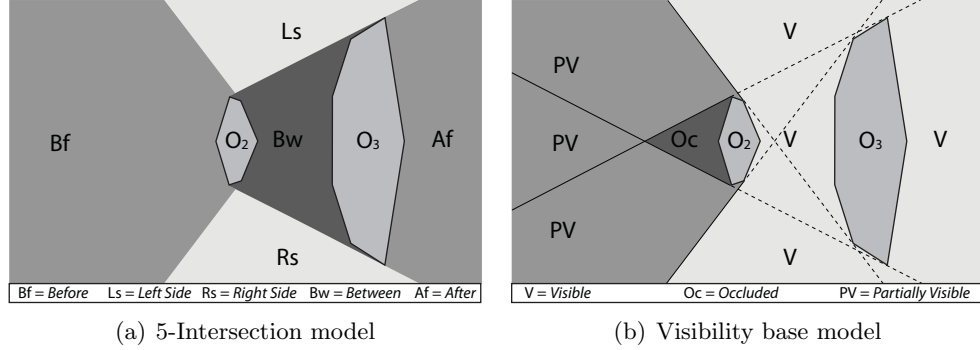


Figure 2.11: Projective ternary calculi. The 5-Intersection model (Billen & Clementini, 2004) defines five projective relations to model relative directions: *Before*, *After*, *Between*, *Left Side*, and *Right Side*. The base visibility model (Tarquini *et al.*, 2007) defines the projective relations of *Visible*, *Occluded*, and *Partially Visible*.

the external and internal mutual tangents between the reference objects. Given three spatial objects O_1, O_2 , and O_3 , the intersection of O_1 with the frame of reference built over the reference objects O_2 and O_3 yields the visibility relation between the three objects. Hence, the Boolean matrix approach used for the CDC calculus can be adopted to represent the visibility relations. Three single-tile relations can hold, corresponding to the symbols V, PV and Oc . Tarquini *et al.* (2007) also describes two permutation operations (namely shortcut and shortcut inverse) and a composition table—defined as $R(O_1, O_3, O_4) = R_1(O_1, O_2, O_3) \circ R_2(O_2, O_3, O_4)$ —for the base single-tile relations of the calculus.

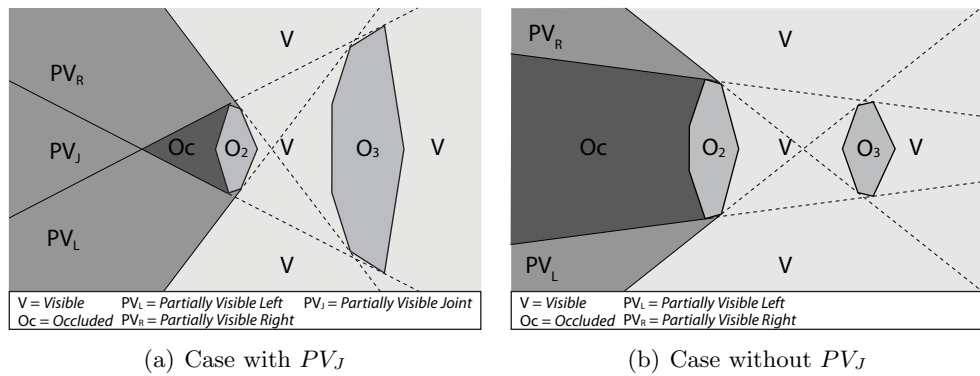


Figure 2.12: Frame of reference of the Visibility Calculus. The Visibility Calculus (Fogliaroni *et al.*, 2009) extends the visibility model by refining the partially visible relation into: *Partially Visible Left* (PV_L), *Partially Visible Right* (PV_R), and *Partially Visible Joint* (PV_J). The acceptance area of PV_J can be either non-empty (a) or empty (b) based on the properties of the reference objects.

Table 2.7: Permutation and rotation of Visibility relations.

R	V	PV_L	PV_J	PV_R	Oc
SC(R)	U _{Vis}	V	V	V	V
SCI(R)	U _{Vis}	V	V	V	V

Table 2.8: Composition of two single-tile visibility relations as in De Felice *et al.* (2010).

R₁ \ R₂	V	PV_L	PV_J	PV_R	Oc
V	U _{Vis}	V, PV _L , Oc	V, PV _L , PV _R , Oc	V, PV _R , Oc	V
PV_L	U _{Vis}	PV _L , Oc	PV _L , Oc	PV _L	PV _L
PV_J	V, PV _L , PV _J , PV _R	PV _R , PV _J	PV _L , PV _J , PV _R , Oc	PV _L , PV _J	PV _J
PV_R	U _{Vis}	PV _R	PV _R , Oc	PV _R , Oc	PV _R
Oc	U _{Vis}	Oc	Oc	Oc	Oc

Fogliaroni *et al.* (2009) refined the Partially Visible relation introducing three new relations that correspond to the concepts of *Partially Visible Left* (PV_L), *Partially Visible Right* (PV_R), and *Partially Visible Joint* (PV_J). The relation $PV_L(O_1, O_2, O_3)$, for example, means that the object O_1 is partly seen from O_3 on the left of O_2 . The acceptance areas of the extended visibility calculus are depicted in Fig. 2.12. As a consequence of its definition, for certain configurations of reference objects, the acceptance area PV_J might not be build; for instance Fig. 2.12(a) shows a configuration of objects for which PV_J exists, while Fig. 2.12(b) depicts a case for which the area does not exist. The set of base single-tile visibility relations is $\mathcal{R}_{Vis}^{ST} = \{V, PV_L, PV_R, PV_J, Oc\}$. Moreover, the set of JEPD visibility relations \mathcal{B}_{Vis} contains 2^5 relations; however, only 22 multi-tile relations are geometrically realizable. For instance, a primary object can not intersect at the same time the areas Oc and PV_J , without crossing also either PV_L or PV_R .

The permutation—shortcut and shortcut inverse—and the composition —defined as $R(O_2, O_3, O_4) = R_1(O_1, O_2, O_3) \circ R_2(O_1, O_3, O_4)$ —operations described in De Felice *et al.* (2010) are shown respectively in Table 2.7 and Table 2.8. However, those tables consider only single-tile base relations.

2.2.4 On the Combination of Qualitative Calculi

The research on QSR has mainly focused on the definition of calculi that represent and reason only on one single aspect of space. However, for real applications, such as the emergency management scenario discussed in Chapter 1, multiple aspects of space interplay among each other. For instance, considering the calculi described in Section 2.2.3, if it is known $PV_J(O_1, O_2, O_3)$, it also holds true that $DC(O_1, O_2)$ and $DC(O_1, O_3)$. The general focus on investigating only a single and isolated aspect of

space at a time is one of the main reasons for the lack of real applications that make use of QSR (Renz & Nebel, 2007).

There exist some works in the QSR field that investigate how the combination of different qualitative spatial aspects of space can be performed. Basically, three different approaches for the combination can be identified in the literature:

COMBINED REASONING. Definition of a reasoning system—namely through the definition of composition algorithms or tables—that combines the relations of two different calculi. As an example, Clementini *et al.* (1997) define some algorithms to compose distance and orientation information, that allow to reason about *positional* information. In contrast, Sharma (1996) defines the reasoning tables to compose topological (Egenhofer, 1991), directional (Frank, 1992), and distance information. In particular, with the concept of *Heterogeneous Spatial Reasoning* he refers to the composition of two relations from two different calculi that returns a relation in one of the two calculi. For instance, a topological relation is composed with a directional relation in order to compute another topological relation. Conversely, with the *Mixed Spatial Reasoning*, two relations of the same calculus are composed in order to compute a relation in a different calculus. For example two topological relations can be composed to compute a directional relation. Finally, the *Integrated Spatial Reasoning* refers to particular composition operations in which any pair of objects is described by more than one qualitative relation. Similarly to the mixed spatial reasoning approach, Guo & Du (2009) discuss how to derive topological relations from two given direction relations.

COMBINED MODELS. Creation of a new qualitative calculus that is grounded on the calculi that model the considered aspects of the space. For instance, Frank (1992) defines a calculus for modeling positional information by combining orientation and distance. Differently, Brageul & Guesgen (2007) create a new model that considers topology, distance and orientation aspects combining the approach—discussed above—of Clementini *et al.* (1997) with a combination of orientation and topology previously introduced in Hernández (1994). Yet, Li (2006) combines the RCC-5 calculus with the DIR9 algebra, that is a coarser version of the RA algebra (Balbiani *et al.*, 1998). Finally, Billen & Kurata (2008) define a new calculus to refine the eight topological relations of the 9I-M considering projective properties of the objects, namely adopting the *Dimensional Model* (Billen *et al.*, 2002).

JOINT SATISFACTION PROBLEM. Investigation of the reasoning properties in the case that more aspects of the space are taken into account. In particular, an extension of the CSP problem is defined, namely the *Joint Satisfaction Problem*. The JSP grounding idea is that, given a set of objects, two qualitative calculi, and for each calculus a constraint network, even if any single constraint network is consistent,

the *joint network* can be inconsistent due to the interdependencies of the relations in the calculi. Hence, the studies on JSP focus on the consistency checking of joint networks. Gerevini & Renz (2002) study the consistency problem for topological relations—using a subset of relations in the RCC-8 calculus that does not include the *PO* relation—and a newly defined calculus to model qualitative size, called *QS*. The work has been used as grounding for Li (2007); Li & Cohn (2009) that investigate on the JSP problem considering topological, size, and also directional (Guesgen, 1989) information. Yet, Liu *et al.* (2009) consider only topological (RCC-8) and directional information, but they focus on the comparison of the JSP problem’s properties when two different directional calculi are considered: the RA algebra and the CDC calculus.

2.2.5 Spatial Information Translation

In this text, the term *translation* of spatial information will be used to highlight those operations that perform a transformation of spatial information from one representation approach—namely quantitative or qualitative—to the other. In particular, the operations that transform from a quantitative representation to a qualitative representation are called *qualification* operations, while the operations that perform a translation from qualitative to quantitative are called *quantification* operations. While there exist some works in the literature that address the qualification problem, quantification operations still represent a challenge (Wolter & Wallgrün, 2012).

Despite the intensional definition of qualitative spatial relations—based on geometric constraints that the objects have to satisfy in order to belong to a certain relation—already gives a mean for the quantification operations, research has focused on the reduction of the computational time complexity of such operations. The operations for the computation of topological relations, and in particular the relations defined in the 9I-M (Egenhofer & Franzosa, 1991), are the only operations to compute qualitative relations between spatial objects that are included in the OpenGIS specifications (Herring, 2001). An algorithm to compute the topological relations between spatial objects has been presented by Schneider (2002). Given two objects O_1 and O_2 , described respectively by n and m vertices, the algorithm performs in $\mathcal{O}((n + m) \log(n + m))$ time in the worst case.

Skiadopoulos *et al.* (2004, 2005) present an algorithm that computes the cardinal direction relations (Goyal & Egenhofer, in press) between two spatial objects O_1 and O_2 that represent spatial regions in *REG**. Being n_1 and n_2 the number of edges of respectively O_1 and O_2 , the proposed algorithm performs in $\mathcal{O}(n_1 + n_2)$ time in the worst case.

Finally, Clementini & Billen (2006) describe an algorithm to compute the relations for the 5-intersection model (Billen & Clementini, 2004) through the computation of

the internal and external mutual tangents between the reference spatial objects. Given three spatial objects O_1, O_2 and O_3 , having respectively n_1, n_2 , and n_3 vertices, the algorithm performs in $\mathcal{O}(n \log n)$ time in the worst case, where $n \triangleq n_1 + n_2 + n_3$.

2.3 Uncertain Spatial Knowledge

Representation of spatial knowledge has traditionally been concerned with the assumptions that spatial regions have well-defined boundaries and that they can be adequately observed and represented as *sharp* spatial objects (Pauly & Schneider, 2008): vector and half-plane representations (Section 2.1.1) as well as qualitative models (Section 2.2.3) are suited for such kinds of entities. However, the assumptions turn out to be inappropriate since *certain* knowledge is only available in abstract domains, while in real domain knowledge is subject to *uncertainty* due to limitations in knowledge acquisition and representation (Freksa, 1994). In the remainder of this text, the term *uncertain region* will be used to denote a spatial region which description is uncertain.

Two classes of *uncertainty* in spatial knowledge are defined by Robinson & Frank (1985): The first is related to the inability of exactly observe and represent characteristics of spatial entities where the characteristics are inherently exact, while the second kind of uncertainty derives from intrinsic ambiguity in the concept to represent.

In the first case, uncertainty is mainly inherited from data collection and from the limitations of the used representation approaches. It is grounded on the fact that knowledge about reality is collected by means of observations; any observation and representation is uncertain since it is not capable to correctly reflect all aspects of reality (Duckham *et al.*, 2001). In order to avoid ambiguities in the terminology, the different kinds of uncertainty are defined here consistently with the *Metrology* definitions given in BIPM *et al.* (2008). *Precision* refers to information expressed as a range of possible values (Altman, 1994). Precision is often numerically represented by measures of imprecision, like the *coefficient of variation*. As an example, considering a spatial region whose maximum length is $10m$ (true value), the measure $10m \pm 0.01m$ is more precise than the measure $10m \pm 0.10m$. In contrast, *accuracy* is the degree of closeness of an observation or representation of a quantity to that quantity's actual true value (e.g., Worboys, 1998a). For example, considering again a spatial region having maximum length equals to $10m$ (true value), the measures $10m \pm 0.01m$ and $10.1m \pm 0.01m$ have the same precision, but the former is more accurate than the latter. Besides, the smallest change in the true value that causes a perceptible change in the observation or representation is called *resolution* or *granularity*¹. For instance, the values $10m$ and

¹Since *resolution* and *granularity* affect *precision*, the terms are often used in the literature as synonyms for the same concept (e.g., Duckham *et al.*, 2001; Worboys, 1998b). However, in this thesis the concepts are distinguished in order to highlight the different kinds of uncertainty.

10.00m represent the same true value at different resolutions. Finally, *incompleteness* is related to lack of relevant information to describe a spatial entity (Worboys, 1998b).

The second class of uncertainty defined by Robinson & Frank (1985) deals with those spatial regions which boundaries can not be exactly defined; this aspect of uncertainty is usually called *vagueness*. Two different theories of vagueness are discussed in the literature that distinguish between linguistic and ontological nature of vagueness (see Varzi, 2006). In the first theory, called *de dicto* view, vagueness is a property of names and predicates that describe an entity, and it is not a property of the entity itself (e.g., Bitner & Smith, 2003; Varzi, 2001; Zadeh, 1979, 1995). For instance, Zadeh (1979, 1995) distinguish among *fuzzy proposition*—that is a proposition containing fuzzy terms—, and *vague proposition*—that is (i) a fuzzy proposition and (ii) insufficiently specific for a specified purpose. The second theory of vagueness, also called *de re* view, considers vague spatial entities as entities having truly fuzzy boundaries (e.g., Duckham *et al.*, 2001; Erwig & Schneider, 1997; Fisher, 2000; Tye, 1990). Fisher (2000) proposes the philosophical *Sorites paradox* (Cargile, 1969) to test whether a geographic concept is vague.

2.3.1 Modeling Uncertain Spatial Knowledge

Different approaches have been proposed to model uncertain spatial knowledge. Those approaches can be classified into four main categories:

FUZZY SETS. Firstly introduced by Zadeh (1965), they are based on an extension of the *set* concept: given a set S with the elements in a domain \mathcal{D} , any elements of \mathcal{D} can either belong or not belong to S . Differently, in fuzzy sets a real number, varying between 0 and 1, is assigned to any member of the domain \mathcal{D} : this value indicates the degree of membership of the element to the set. Research in the application of fuzzy sets for describing uncertain regions have been described by Altman (1994); Dilo *et al.* (2007); Guesgen & Albrecht (2000); Liu & Shi (2006); Schneider (1999); Zhan (1998). However, one of the main drawbacks related to fuzzy sets consists in the assignment of the membership values to any element of the domain (Keefe, 2000). A simplified version of fuzzy sets, that does not require the explicit definition of the membership values, has been used in Freksa (1980, 1982); Freksa & López de Mántaras (1982).

THREE-VALUED LOGICS. They have a similar approach to fuzzy sets, but they admit only three distinct degrees of membership: *Yes*, *No*, and *Maybe*. Approaches to model uncertain regions using three-valued logic have been proposed in Cohn & Gotts (1996); Lehmann & Cohn (1994)—*egg-yolk* approach—, Clementini & Di Felice (1996, 1997a, 2001)—*broad boundary* approach—, and Erwig & Schneider (1997); Pauly & Schneider (2004, 2008)—*vague regions* approach. In general,

three-valued logic models define two regions (one containing the other) to model uncertain spatial knowledge: The internal region—called *yolk* in the egg-yolk approach, *inner region* in the broad boundary approach, and *kernel* in the vague regions approach—clusters the points that surely belong to the uncertain region. Conversely, the points that are in the external region—respectively called *egg*, *outer region*, and *boundary*—but are not in the internal one, constitute the set of points that may belong to the uncertain region. All other points surely do not belong to it.

ROUGH SETS. Initially described by Pawlak (1982), they are similar to the three-valued logics approaches, but they rather make use of discrete partitions of the space. Two *approximations* are identified: the *lower* approximation composed by those elements of the partitions that surely belong to the uncertain region, and the *upper* approximation composed by the elements that belong or may belong to the uncertain region. Approaches to model uncertain spatial knowledge using the rough set theory have been proposed in Beaubouef *et al.* (2007); Bittner & Stell (2001); Worboys (1998b).

PROBABILISTIC METHODS. They use statistical and probabilistic functions to define the probability of a certain entity to be located in a specific position. Such methods have been used in Li *et al.* (2007); Mark & Csillag (1989); Tøssebro & Nygård (2008).

2.3.2 Uncertain Spatial Knowledge in GIS

Different approaches have been proposed to represent uncertain quantitative spatial knowledge in Geographic Information Systems, exploiting either raster or vector representation methods already implemented in existing GIS.

Fonte & Lodwick (2005) introduce a representation method based on space tessellation (raster). They define the so-called *fuzzy geographical entity*, that represents an uncertain region by assigning a grade of membership to the region for each element of the tessellation, using the fuzzy set approach. Fuzzy sets have been used also in Erwig & Schneider (1997); Schneider (1999). However, differently from Fonte & Lodwick (2005), they extend the vector representation approach to represent *fuzzy points*, *lines*, and *polygons*.

A three-valued logic representation has been discussed by Clementini & Di Felice (2001). They propose an extension of the vector representation to integrate regions with a broad boundary in the datatype definition given by the OpenGIS specifications for SQL. Single-region and multi-region objects with a broad boundary are defined as counterpart for the single-region and multi-region objects with a sharp boundary. Furthermore, beyond the investigation on topological relations, Clementini & Di Felice (1997a) define spatial operators for regions with a broad boundary, such as Buffer zones, MBR, and Convex Hull. Similarly, Pauly & Schneider (2004) define basic datatypes

for uncertain points, lines, and regions as well as set operators by adopting the vague region approach.

Vector representation extensions that use probabilistic models have been proposed by Dutton (1992); Tøssebro & Nygård (2008). While Dutton (1992) only defines points and lines, Tøssebro & Nygård (2008) introduce all vector datatypes, defining *uncertain lines, points, and regions* using the probabilistic value for each point in the plane to be part of the uncertain region. Also simple operators for such datatypes are defined, such as the intersection operation.

2.3.3 Qualitative Relations Between Uncertain Regions

Besides the definition of what an uncertain region is, and how it is possible to model uncertainty, different research investigated how to model qualitative relations holding between uncertain regions. Those relations are usually called *fuzzy relations* or *approximate relations* (e.g., Clementini & Di Felice, 1997a; Du & Guo, 2010; Schockaert *et al.*, 2006; Zhu *et al.*, 2010), to highlight the contrast with the crisp relations between certain regions.

Most of the existing literature focuses only on the topological aspects of uncertain regions. In particular, RCC-8 relations have been discussed by Cohn & Gotts (1996); Roy & Stell (2001)—using the egg-yolk approach—, by Li & Li (2004); Schockaert *et al.* (2006)—that uses the fuzzy set approach—, and by Bittner & Stell (2001)—considering rough sets. Yet, the 9I-M has been investigated by Clementini & Di Felice (1996, 1997a, 2001) using the broad boundary approach and by Zhan (1998) with the fuzzy set approach. Beaubouef *et al.* (2007) use the rough set model to define topology both as RCC-8 relations and as 9-IM relations. Finally, other approaches make use of ad-hoc definitions of topological relations using either fuzzy set approaches (Dilo *et al.*, 2007; Liu & Shi, 2006; Schneider, 1999) or three-valued logic approaches (Erwig & Schneider, 1997; Pauly & Schneider, 2004, 2008).

Few works investigate how qualitative aspects of the space—different from the topological one—can be modeled when the regions are uncertain. In particular, Cicerone & Di Felice (2000); Du & Guo (2010) investigate how the CDC calculus can be extended to consider regions with a broad boundary as primitive for the model. Furthermore, Zhu *et al.* (2010) investigate an algorithm to compose approximate cardinal direction relations, extending the composition algorithm proposed by Skiadopoulos & Koubarakis (2004).

Chapter 3

Spatial Information Integration for Emergency Response

The state of the art with respect to spatial information collection, representation, and manipulation has been analyzed in the previous chapter. The present chapter discusses how the described techniques relate to the emergency management scenario introduced in Chapter 1. Section 3.1 analyzes the requirements of spatial information after extreme events. Limits in the methods for collecting spatial data after extreme events will be discussed considering as an example the earthquake that struck the Haiti's capital city of Port-au-Prince in January 12, 2010. An alternative approach for spatial data collections is proposed that exploits communications exchanged among the extreme event responders. However, communications convey spatial information that is *qualitative* in nature, while GIS requires *quantitative* spatial information. The integration of existing information with information extracted from the communication is crucial for providing responders with updated descriptions of the environment. Thus, a system for the integration of qualitative and quantitative spatial information is proposed in Section 3.2. Challenges addressed in this thesis that are related to the integration of mixed descriptions of spatial knowledge will be detailed in Section 3.3.

3.1 Geographic Data Collection for Emergency Response

Typically, *Geographic Information Systems* (GIS) store and manage detailed spatial information collected by national or local administrations, industries, military forces, etc. Each of them stores in a GIS all the required information for their particular goals: a local administration needs the descriptions of roads, buildings, and infrastructures; an industry enterprise collects information useful for its kind of business; military forces

store and manage information related to strategic points or risk zones within an urban environment.

Spatial information changes are often slow processes and the changes affect only a minimal part of the whole information stored in a GIS. As an example, the process of construction of a new road or building requires a long time to be completed, and it affects only the neighbor of the new entity. In this case, the operations required to update the GIS can be straightforwardly defined: accurate measures, using for instance *ground surveying* or *GPS* methods, of the new entity are needed, and the new information has to be added into the system. Information related to neighboring entities affected by the change can be updated using the same approach.

This scenario drastically changes when an unexpected natural event, like an earthquake or a tornado, modifies the typical *static* environment in few seconds (cf. Section 1.2.2). This kind of event usually causes a lot of changes in the environment: buildings and bridges can collapse, landslides can occlude roads, etc. Conversely from the situation described before, information changes are not slow processes anymore and they can affect a big part of the whole information managed by GIS. Obviously these kind of changes make the environment description stored in a GIS unreliable; if the information is out of date, all the advantages of using GIS are lost. At the same time, after a natural catastrophe, GIS information is particularly important as it is needed to coordinate the different aid operations.

In this section, the limits in data collection after a natural disaster will be discussed.

3.1.1 Emergency Management

Emergency Management (EM) has been defined as the “*discipline and profession of applying science, technology, planning and management to deal with extreme events that can injure or kill large numbers of people, do extensive damage to property, and disrupt community life*” (Drabek & Hoetmer, 1991). The EM field has been deeply investigated by different communities, such as Computer Science, Environmental Science, Engineering, and Business. Extreme events are classified as *Natural* (e.g., earthquakes, tsunami, eruptions of volcanoes, etc.) or *Human Driven* (e.g., nuclear emergencies, terrorism, etc.). Four different phases are defined for emergency management, based on the *Comprehensive emergency management model* firstly introduced in National Governors’ Association (1978):

MITIGATION. Preventing future emergencies or minimizing their effects. It includes any activity that prevents an emergency, reduce the chance of an emergency happening, or reduce the damaging effects of unavoidable emergencies. Mitigation activities take place before and after extreme events.

PREPAREDNESS. Preparing to handle an emergency. It includes plans or preparations to save lives and to help response and rescue operations. Preparedness activities take place before an extreme event occurs.

RESPONSE. Responding safely to an emergency. It includes actions taken to save lives and prevent further property damages in an emergency situation. Response is putting preparedness plans into action.

RECOVERY. Recovering from an emergency. It includes actions taken to return to a normal or an even safer situation following an emergency. Recovery activities take place after an extreme event.

The capability of a community to respond to and to recover from an extreme event is called *resilience*. Renschler *et al.* (2010) identify seven different dimensions of resilience (*PEOPLES resilience framework*—Fig. 3.1(a)) that consider both physical/environmental and socio-economical aspects of a community. The resilience of a community is strongly influenced from the actions taken during all four EM phases as well as from the availability of information (spatial as well as non-spatial) related to any dimension of the framework. The work presented in this thesis focuses on the *Physical infrastructure* dimension of the PEOPLES framework.

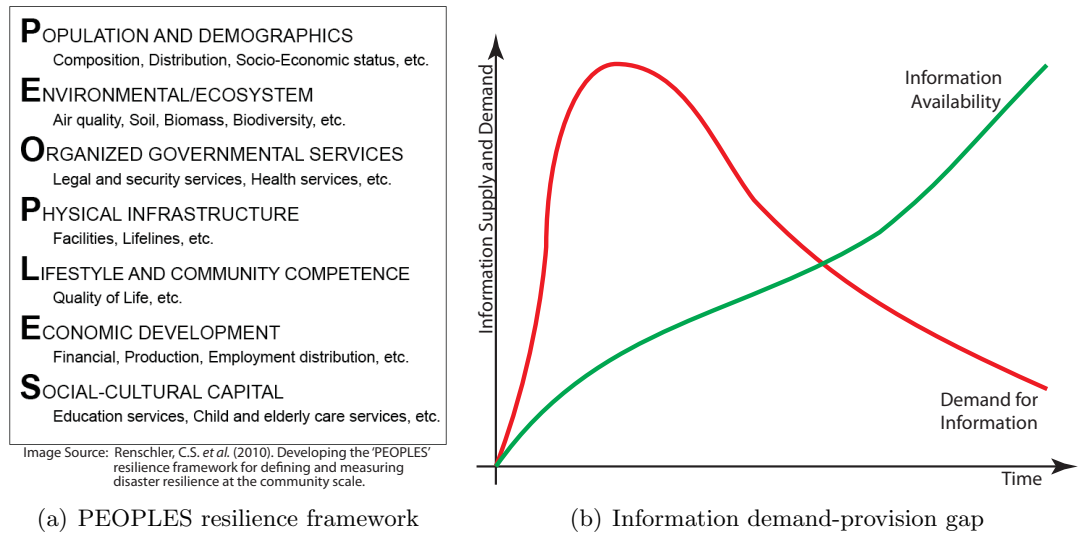


Figure 3.1: *PEOPLES resilience framework* (Renschler *et al.*, 2010) and the information gap after an extreme event (MacFarlane, 2005).

Being any environment dynamic in nature, imagery and geographic data always represent the environment only at one point in time. However, during the mitigation, preparedness, and recovery phases there is, in general, a low chance that information required to reach the phase's goals is outdated (i.e., in the *physical infrastructure* dimension the description of a road or building stays unvaried for a considerably long

time). Furthermore, the collection of any kind of spatial data required to reach the goal is not a time critical operation. The same consideration does not hold true for the response phase. Indeed, during an extreme event there exists a high chance that there is a drastic change in geographic information at any of the seven resilience dimensions, and continuous data and imagery collection becomes crucial. Nevertheless, the time critical goals to achieve during this phase makes the collection of updated spatial information a big challenge. This problem has been discussed in MacFarlane (2005), that shows how the demand of information after an extreme event increases faster than that of supply, leading to the *demand-provision gap* shown in Figure 3.1(b). The gap needs to be fulfilled by collecting updated information; for a successful response it is also crucial to share the updated information with all actors involved in the response phase's tasks.

3.1.2 Example: Information Availability after the Haiti Earthquake

Traditional methods for collecting geographic data have been introduced in Section 2.1.3. However, those methods fail in providing updated information rapidly enough to support the response phase operations. Indeed, most of these operations (such as rescue of victims and settlement of gathering points) are performed in the first few hours following the extreme event, while traditional methods for data collection require more time to gather and provide updated information.

In this work, the magnitude 7.0 earthquake that struck the Haiti's capital city of Port-au-Prince in January 12, 2010¹ is considered as a reference for analyzing the information availability after an extreme event. The *MCEER Earthquake Engineering to Extreme Events* center² at the University at Buffalo - The State University of New York (SUNY) provides an overview of the data collected after the earthquake through its *Global Disaster Database*³. The database documents the different geographic data providers that collected data after the extreme event struck the city. The supplied data can be clustered according to the primary and secondary methods classification introduced in Section 2.1.3.

Data collected after the earthquake through primary collection methods⁴ is summarized in Table 3.1. As the table shows, low resolution satellite imagery have been available starting from the day after the catastrophe, while high resolution data required more days to be collected. While satellite systems only need a different configuration of their parameters to point on a specific zone of the earth surface, the other primary collection methods require the settlement of complex devices on the ground. Thus, aerial

¹USGS Report: <http://earthquake.usgs.gov/earthquakes/recenteqsww/Quakes/us2010rja6.php>

²MCEER, Earthquake Engineering to Extreme Events: <http://mceer.buffalo.edu/>

³MCEER Global Disaster Database: <http://mceer.buffalo.edu/infoservice/databases.asp>

⁴Survey and GPS measurements, satellite images and aerial photographs (cf. Section 2.1.3.1).

Table 3.1: Haiti Earthquake: raster data availability.

DATA AVAILABILITY	SOURCE	COLLECTION METHOD
13-Jan-2010	DigitalGlobe ^a	Satellite imagery
13-Jan-2010	GeoEye ^b	Satellite imagery (Low resolution)
15-Jan-2010	UNOSAT ^c	Satellite imagery
17-Jan-2010	GeoEye ^b	Satellite imagery (High resolution)
27-Jan-2010	IPLER ^d	Aerial LIDAR DEM, SEM, point clouds
27-Jan-2010	IPLER ^d	Aerial 3D imagery
05-Feb-2010	AIDG ^e	Ground data

^a http://www.digitalglobe.com/content/haiti/haiti_viewer.html^b <http://www.geoeye.com/>^c <http://www.unitar.org/unosat/maps/49>^d <http://ipler.cis.rit.edu/projects/haiti>^e <http://www.aidg.org/>**Table 3.2:** Haiti Earthquake: vector data availability.

DATA AVAILABILITY	DATA SOURCE	SOURCE	SUBJECT
14-Jan-2010	Satellite	SERTIT ^a	Damage assessment map
14-Jan-2010		HIU ^b	Landslide hazards
15-Jan-2010	Satellite	SERTIT ^a	Spontaneous gathering areas
15-Jan-2010	Satellite	SERTIT ^a	Visibly damaged buildings
15-Jan-2010	UNOSAT	UNITAR ^c	IDP concentrations, road and bridge obstacles
16-Jan-2010	Satellite	SERTIT ^a	Location of visible water surfaces
16-Jan-2010	UNOSAT	UNITAR ^c	Damage assessment for major buildings/infrastructures
18-Jan-2010	UNOSAT	UNITAR ^c	Density of bridge and road obstacles
4-Feb-2010	UNOSAT	UNITAR ^c	Comprehensive building damage assessment

^a http://sertit.u-strasbg.fr/SITE_RMS/2010/01_rms_haiti_2010/01_rms_haiti_2010.html^b <https://hiu.state.gov/>^c <http://www.unitar.org/unosat/maps/49>

pictures and ground data collection needed more time than satellites imagery (almost two weeks from the disaster). However, satellite as well as aerial pictures only provide information in a raster format (a certain value of the measured parameter correspond to any pixel of the image). Thus, this data only gives a visual overview of the damaged areas, but does not provide exact and structured information as needed in the response phase (e.g., a list of the collapsed building where there is a high probability to find victims or a list of the blocked roads).

The imagery are analyzed using secondary methods¹ of data collection, that allow for the extraction of vector data from the pictures. The interpretation of the imagery requires the execution of computationally expensive algorithms of image analysis. A synthesis of spatial information available in vector format after the Haiti event is reported in Table 3.2. The table shows that a damage assessment map has been available two days after the event, followed by other kind of information in the days after.

The summary of the data collected after the Haiti earthquake (both primary and secondary methods) is depicted in Figure 3.2. The figure clearly shows the information provision lack (discussed in the previous section) following the event. The first updated information was available the day after the event, and its format is not suitable for most of the response tasks, that rather need information in vector format. For instance, the knowledge about the functionality of infrastructures can improve path planning and navigation of the emergency vehicles involved in the rescue operations, while knowledge about the damaged areas, cross-checked with the population density of each area, can improve the planning of aid operations. Both examples show critical operations that are performed as soon as the event struck.

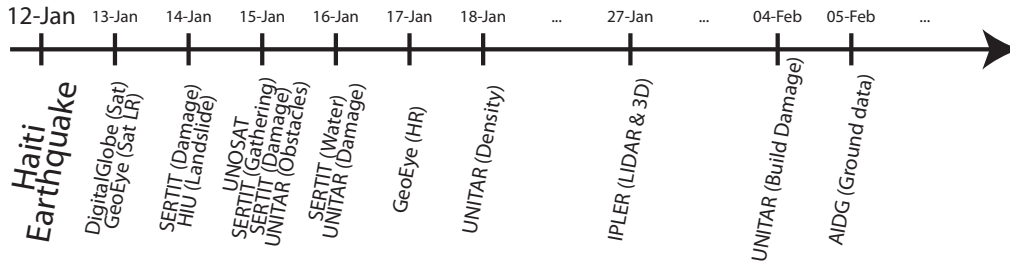


Figure 3.2: Haiti earthquake.

The consequence of the lack of information is that all the operation performed by search and rescue missions after the event have been based on out-of-date descriptions of the environment. Due to their time-critical characteristics, updated spatial information

¹Scanners, manual and head-up vectorization, photogrammetry (cf. Section 2.1.3.1). Scanner and manual or head-up vectorization are not suitable to collect data after extreme events since they are grounded on already existing maps. Hence, only photogrammetry is a suitable technology to extract vector data from photographs and imagery.

can make the difference between a successful or unsuccessful accomplishment of the response phase operations.

3.1.3 Information Lack in the EM Response Phase

The lack of information shown in the case of the Haiti earthquake can be generalized for all natural extreme events that strike a populated region. As Figure 3.3 shows, the spatial information is up-to-date until the event happens. After a certain time from the extreme event, information is again up-to-date since new data is collected using traditional methods, either primary or secondary. The amount of time strongly depends on the kind of information required to solve a particular task: a preliminary evaluation of the criticism can be done with the low resolution data gathered from satellites, while a detailed analysis of the infrastructures requires more precise vector data. However, independently of the kind of information, there always exists a *lack of updated information* going from when the extreme event struck to when the updated information is available. As shown for the Haiti earthquake, traditional methods of data collection fail in providing useful information during such interval and alternative kind of information sources need to be taken into consideration.

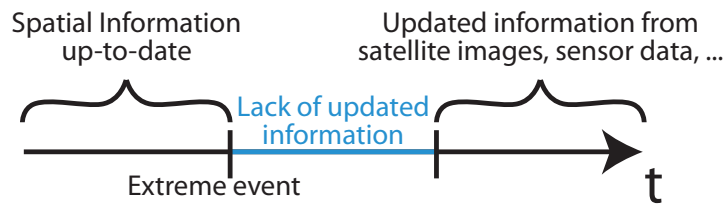


Figure 3.3: Extreme event: Lack of information in the response phase.

3.1.4 VGI for Emergency Management

The necessity to exploit the Internet and GIS technology to collect and share data to support the response operations after an extreme event has been firstly discussed by Goodchild (2003). Afterwards, in Goodchild & Glennon (2010) VGI technologies—closely related to *crowdsourcing*¹—have been proposed as means of collecting data from volunteers, focusing especially on issues related to data quality and trustability. This led, in recent years, to the development of social networks for supporting data collection in emergency management (Li & Goodchild, 2010).

¹According to Goodchild & Glennon (2010), the term crowdsourcing (Howe, 2008) has mainly two meanings: (1) “it can refer to the proposition that a group can solve a problem more effectively than an expert, despite the group’s lack of relevant expertise”, and (2) “information obtained from a crowd of many observers is likely to be closer to the truth than information obtained from one observer”.

According to this idea, diverse social network-based systems for collecting geographic data have been employed after the earthquake in Haiti, among which CrisisCamp Haiti¹, OpenStreetMap, USHAHIDI², and GeoCommons³ (Zook *et al.*, 2010). The first two services provided volunteers with web-based platforms for on-line mapping to create updated descriptions of the infrastructures (e.g., buildings and roads); those services relied on satellite imagery furnished by GeoEye, and hence available only one day after the catastrophe (cf. Table 3.1). Differently, USHAHIDI (Okolloh, 2008) allows users to submit reports either by using the most diffused social networks (e.g., Twitter), or by sending SMS or MMS. Those reports are *geo-tagged*—meaning that a position is associated in a map to any text report—and then they are made available for decision makers. Finally, GeoCommons provides different web tools to integrate and query maps and imagery shared by different authorities; also this service is dependent on data collected using traditional methods.

Even though the described services have successfully provided volunteers with platforms to collect geographic data for supporting the emergency management response phase, still the proposed approaches do not compensate the information lack discussed in Section 3.1.3. Indeed, the social network-based collecting methods require the users to expressly connect to the offered web service. Such a requirement represents a strong limiting factor for people affected by the event and for rescuers involved in the response operation in sharing their knowledge through the system. The former have as main priority to reach a gathering point to receive assistance, and, even though they would have the chance to share information, they can not rely on existing communication infrastructures that are often heavily damaged by extreme events. The latter can usually make use of *ad-hoc* communication infrastructures, yet their priority is to save and rescue people affected by the event; this duty takes them away from the intended users of the social network systems.

In this work, an innovative system is described that exploits the interpretation of the communications among the actors involved in the response phase—both people affected by the event and rescuers—for updating the existing geographic knowledge. As shown in the example of Chapter 1, such communications convey a large amount of spatial information that is not yet exploited in existing VGI systems.

3.2 A Geographic Information Integration System

Reports and communication exchanged among disaster responders, as well as calls to the emergency lines, can be exploited to update the geographic knowledge of a pop-

¹<http://crisiscommons.org/>

²<http://haiti.ushahidi.com/>

³<http://geocommons.com/>

ulated environment. Existing limits in the VGI applications developed for emergency management can be removed by wiretapping the communication exchanged among the different actors involved in the disaster, and opportunely interpreting, storing, and sharing them.

3.2.1 Spatial Information Extraction and Representation

Communications among rescuers as well as distress calls received by the emergency lines have to be interpreted in order to extract the spatial information they convey. At first, the described spatial entities are extracted out of them. Two kind of entities are identified:

KNOWN ENTITIES. They refer to elements in the environment that were existing also before the event and did not undergo any change. These entities will be represented by *known objects* (denoted as O_1, O_2, \dots) since their quantitative description is known and precise.

UNKNOWN ENTITIES. They refer to the spatial entities that have been created as a consequence of the extreme event (such as a fire) as well as elements that did undergo a change after the event (for which the eventually available quantitative description is out-of-date). They are represented by *unknown objects* (denoted as O_1^*, O_2^*, \dots) since their extension and position is not precisely known but can only be approximately identified by interpreting other available information.

In addition to known and unknown entities, information that relates them are extracted from the communication. Indeed, as shown in Section 2.2, humans use qualitative expression to manage spatial information, rather than quantitative approaches that convey numerical information as done by GIS systems. Hence, the interpretation of the reports yields a set of qualitative information that relates known and unknown entities. For the purpose of this work, only three spatial aspects will be taken into account: topology, that well describes containment relations (e.g., the storage site is *inside* the hazardous area), cardinal directions (e.g., the fire is *north-west* of the airport), and information reporting the possibility to see a certain entity from a given position (e.g., I can *see* the fire from the college). However, the procedures that will be explained in this thesis can be straightforwardly applied to other aspects of the space. The considered calculi to model qualitative information are hence the RCC-8 (Section 2.2.3.1), the CDC (Section 2.2.3.2), and the Visibility calculus (Section 2.2.3.3).

A strategy to integrate qualitative and quantitative spatial information is developed, that provides methods to approximate the spatial extension of unknown entities—resulting in the possibility of drawing them on a map—and to derive new qualitative information useful for querying purposes (e.g., to retrieve the dangerous entities *inside*

a hazardous area). It is not in the scope of this thesis to argue whether qualitative information is more suited than quantitative information in emergency situations; rather the target is to show how it is possible to integrate both approaches in order to exploit any kind of information available after an extreme event¹.

3.2.1.1 Assumptions

Issues related to the communication collection and information extraction are not analyzed in this work. At first, it is assumed that a system to collect and store all the communications in a shared repository, and to efficiently retrieve them, exists. Indeed, a reliable communication infrastructure has to be established that allows the rescuers to communicate their reports. Existing infrastructures, such as the ones used for mobile communication, are not reliable anymore since they are usually heavily damaged by extreme events. Rather, an *ad-hoc* infrastructure, like the ones used by military forces and fire brigades, furnishes a more reliable basis for the communications.

In addition, it is not in the objective of this work to investigate how natural language can be interpreted, rather the focus is on how the extracted information can be transformed to perform updates in the existing geographic knowledge. Hence, it is supposed that known and unknown entities are automatically extracted and interpreted from the communications. Furthermore, same entities that are reported in two or more reports are automatically associated with the same symbol. For instance, if a fire is reported and associated in one interpretation with the symbol O_F^* , the same symbol will be used to refer to the fire if it also appears in other reports. Moreover, it is assumed that all the extracted information is consistent.

Finally, even though the events that follow a natural disaster usually have a dynamic evolution, only a static environment is considered in this work. Hence, if an entity has been reported in a certain area, it is assumed that it does not change its position at a later time. Similarly, issues related to information persistence (i.e., for how long a certain information is considered to be valid) are not addressed in this work.

3.2.2 Spatial Information Integration

A simple example², that shows how qualitative and quantitative information pieces are integrated with each other, is depicted in Fig. 3.4. The coordinates of the river (O_{RI}), the airport (O_{AI}), the university college (O_{UC}), the hazardous material storage site (O_{HM}), and the actual position of the ambulance (O_{AM}) are known. Hence, those objects quantitatively describe the set of known entities. Conversely, the fire area

¹For a discussion about the level of abstraction required for specific applications, see (Freksa, 1991).

²The example shown here is a schematization of the example described in Chapter 1 (Fig. 1.2, p. 3).

(depicted as the *Hazardous Area* HA and represented by the imprecise object O_{HA}^*) is the only element that composes the set of unknown entities.

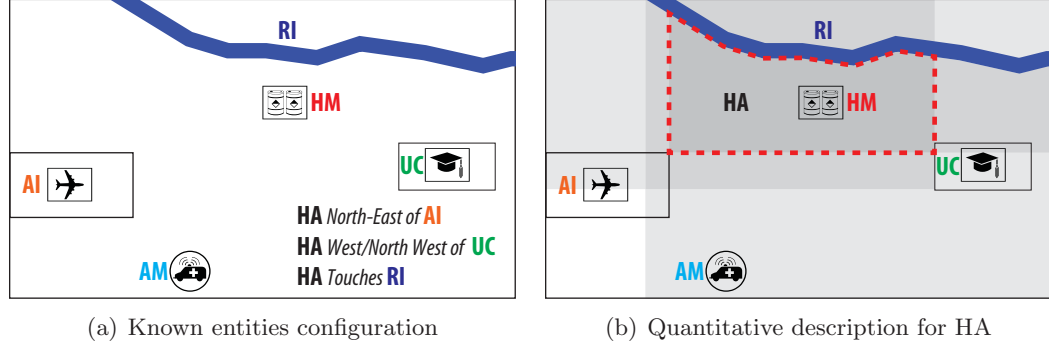


Figure 3.4: Inference of a quantitative approximation for the Hazardous Area (HA).

As shown in Chapter 1, it is supposed that the fire is described in the communications received by different emergency lines¹. Qualitative spatial relations are extracted from the reports as shown in Fig. 3.4(a). For instance, $NE(O_{HA}^*, O_{AI})$ and $\{W, NW\}(O_{HA}^*, O_{UC})$ are qualitative information pieces that are extracted from the communications and that relate the entities in Fig. 3.4.

Even though the available information is represented in a formal way (either qualitatively or quantitatively), its expressiveness is not fully exploited yet. To do so, the first step is to quantitatively interpret the qualitative descriptions of O_{HA}^* in order to identify which area is potentially affected by the fire. The area can be reduced by opportunely combining the interpretations of different qualitative information pieces (Fig. 3.4(b)). Obviously, the inferred description is only an approximation of the real spatial extent of the hazardous area, since it comes from the interpretation of information that only partially describes it. However, even if its description is imprecise, the hazardous area can be shown on a map, providing the EM's decision makers with a visual overview of the dangerous areas. Furthermore, the extracted quantitative description is integrated with information previously available in order to discover new qualitative information, that can eventually trigger rapid warning messages to the decision makers. For instance, O_{HM} turns to be into the hazardous area— $NTPP(O_{HM}, O_{HA}^*)$ —, triggering a warning message for the fire brigades. Finally, a different integration is performed at a purely qualitative level. Indeed, the position of the fire with respect to the airport— $NE(O_{HA}^*, O_{AI})$ —is combined with the position of the airport with respect to the ambulance— $NW(O_{AI}, O_{AM})$. This results in the information that the fire is somewhere north of the ambulance— $\{N, NE, NW\}(O_{HA}^*, O_{AM})$. Thus, the new

¹FIRE BRIGADES: [...] a fire broke out north west of the airport [...]; DISTRESS CALL: [...] There is a big fire in the direction of the Airport but it did not clear the college yet [...] (cf. Section 1.1).

knowledge can be conveyed to an ambulance driver that might decide to search for an alternative path to reach his destination.

3.2.3 System Description

A system for the integration of geographic information is depicted in Fig. 3.5. The system gets as input spatial knowledge already represented either in qualitative or geometric form. The stored information is composed by a set of known entities (quantitatively described by known objects), a set of unknown entities (described by unknown objects), and a list of information that relates entities belonging to the two sets. Information stored in the system can be retrieved to solve different tasks such as *querying* operations that require qualitative information (e.g., searching for all the infrastructures that are *in* a hazardous area), or *visualization* operations that mostly require geometric information (e.g., visualizing a map that depicts the hazardous areas).

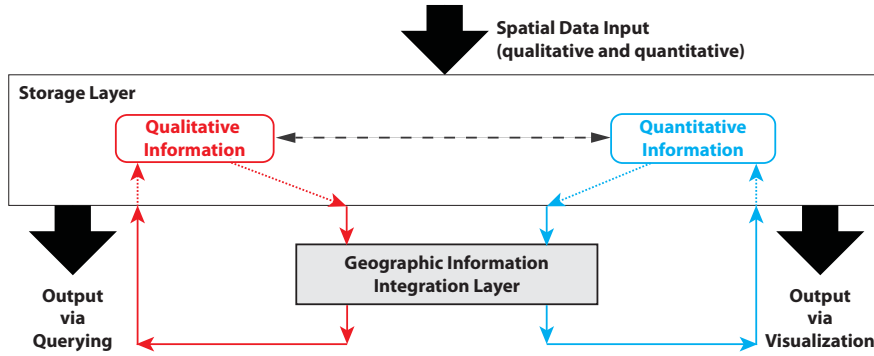


Figure 3.5: Architecture of the geographic information integration system¹ – The *Storage Layer* permanently stores both qualitative and quantitative descriptions of spatial knowledge and opportunely links the information pieces that describe the same spatial entity. It also provides interfaces for information input (qualitative and quantitative) and output (through querying and visualization operations). In contrast, the *Geographic Information Integration Layer* aims at updating the spatial knowledge stored in the storage layer by integrating quantitative and qualitative spatial information.

Two layers establish the core of the system: a *Storage Layer* and a *Geographic Information Integration Layer*. The former simply aims at the permanent storage of qualitative and geometric information, linking every qualitative relation with the corresponding spatial objects it relates. For instance, if the relation $PV_L(O_1, O_2, O_3)$ is given, such relation is represented in the qualitative information storage block, but at the same time the geometric information block stores the geometric information of O_1, O_2 , and O_3 . The storage layer also provides interfaces for spatial information

¹The red lines represent the flow of qualitative information while the cyan lines represent the stream of quantitative information.

input and output. Such layer can be easily seen as a spatial database. Instead, the information integration layer performs the operations necessary for the integration of qualitative and quantitative information. The development of such layer is the focus of this thesis; the arising challenges in the information integration are analyzed in the remainder of this chapter.

3.3 Qualitative and Quantitative Spatial Information Integration

To better understand the challenges related to the integration of spatial information for emergency management, in Section 3.2.2 the example discussed in Chapter 1 has been formalized with respect to the representation methods and the models described in Chapter 2. Subsequently, the architecture of a system to integrate qualitative and quantitative information has been presented in Section 3.2.3. Different aspects and problems to consider when integrating qualitative and quantitative spatial information will be discussed in this section.

3.3.1 Quantification

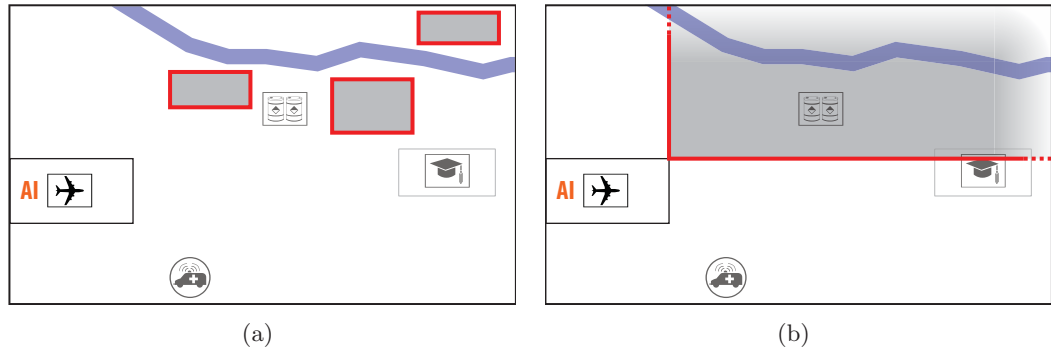


Figure 3.6: Quantification of the relation $NE(O_{HA}^*, O_{AI})$. Three exemplary quantitative descriptions of the relation are shown in (a). The shaded area in (b) is instead the spatial region where HA can lie in order to satisfy the relation with AI .

At first, the operation of *quantification* of qualitative information is considered, that is the translation operation for transforming a qualitative information piece into a quantitative one. Given a qualitative relation $R(O_1^*, O_2, \dots, O_n)$, which primary object O_1^* represents an unknown entity, the quantification of R consists of computing a description of the spatial region that the entity represented by O_1^* can occupy in order to satisfy the given relation with respect to the known reference entities. This means,

it is not sufficient to compute an exemplary description for O_1^* such that it satisfies the relation (e.g., Fig. 3.6(a)), rather it is necessary to identify how the extension of a spatial region that satisfies the relation is quantitatively constrained. For instance, any spatial region that satisfies the relation $NE(O_{HA}^*, O_{AI})$ is contained in the shaded area depicted in Fig. 3.6(b) (corresponding to acceptance area defined in the CDC calculus for the relation NE —cf. Section 2.2.3.2). Thus the extension of the spatial entity can not be bigger than the shaded area. Similarly, any spatial region that satisfies the relation $TPPI(O_{HA}^*, O_{AI})$ contains the spatial region represented by O_{AI} . Thus, the quantification of a qualitative spatial relation is defined by two regions: one represents the upper limit for the extension of O^* , while the other defines its lower limit.

Formally, the operation of quantification of a relation $R(O_1^*, O_2, \dots, O_n)$ consists of identifying the union of all the regions in \mathbb{R}^2 that, standing in for O_1^* , satisfy R —defining a spatial object that describes the spatial region containing all the spatial regions that satisfy the relation—, and of the intersection of all the regions in \mathbb{R}^2 that, standing in for O_1^* , satisfy R —that identifies an object that describes the region contained in all the spatial regions that satisfy the relation (Definition 9). The former object will be denoted as $A_R^+(O_2, \dots, O_n)$, while the latter as $A_R^-(O_2, \dots, O_n)$. The object $A_R^+(O_2, \dots, O_n)$ (called *maximal quantification*) represents by definition the *maximal extension* that the unknown entity represented by O_1^* can have in order to satisfy the relation $R(O_1^*, O_2, \dots, O_n)$. Similarly, $A_R^-(O_2, \dots, O_n)$ (called *minimal quantification*) constraints the *minimal extension* of the spatial entity represented by O_1^* . For the sake of simplicity, in this text $A_R^+(O_2, \dots, O_n)$ (respectively $A_R^-(O_2, \dots, O_n)$) will be denoted only with A_R^+ (respectively A_R^-) when the set of reference objects is clear from the context.

Definition 9 (Quantification of a qualitative relation).

Given a qualitative relation $R(O_1^*, O_2, \dots, O_n)$ defined in an n -ary calculus C , let O_1^* be an unknown object, and let O_2, \dots, O_n be known objects, the quantification of R is composed of a pair of objects defined as:

$$A_R^+(O_2, \dots, O_n) \triangleq \bigcup_{\substack{O \subseteq \mathbb{R}^2 \\ R(O, O_2, \dots, O_n)}} O \quad A_R^-(O_2, \dots, O_n) \triangleq \bigcap_{\substack{O \subseteq \mathbb{R}^2 \\ R(O, O_2, \dots, O_n)}} O$$

Of course, a relation $R(O_1^*, O_2, \dots, O_n)$ can be quantified if and only if all the reference objects have an associated geometry in the system, otherwise it is not possible to build the frame of reference for the relation. Furthermore, based on its definition, the quantification of a relation is totally independent of the primary object; considering for instance two unknown objects O_A^* and O_B^* for which two cardinal direction relations $N(O_A^*, O_2)$ and $N(O_B^*, O_2)$ exist in the system, the quantification of the two relations

produces exactly the same result. Note that the object A_R^+ resulting from the quantification of a single-tile base projective relation (CDC and VC) corresponds exactly with the representation of the acceptance area defined for the relation.

3.3.2 Spatial Regions with Infinite Extent

The quantification operation often is required to compute objects that represent spatial regions with *infinite extent*. For instance, considering again the relation $NE(O_{HA}^*, O_{AI})$, the quantification of the relation yields the shaded area in Fig. 3.6(b); the area has a well-defined boundary at its bottom and left sides, while it extends into infinity on its top and right sides. A regions with infinite extent has, in general, a well-defined *sharp boundary* and one or more *infinite boundaries* that extend into infinity.

The vector representation method is not able to represent this kind of regions, since it requires the exact definition of the boundaries of any spatial region. The representation of the only finite boundary would in fact be indistinguishable from the definition of a simple line string. A straightforward solution to this problem consists in the definition of an *area of interest*¹ for a specific application. Thus, the quantification result is limited by the defined area of interest and it is always a region with finite extent. For instance, in the discussed earthquake scenario, the area of interest might be defined as the administrative boundaries of the city struck by the extreme event (e.g., the black box in Fig. 3.6(b)).

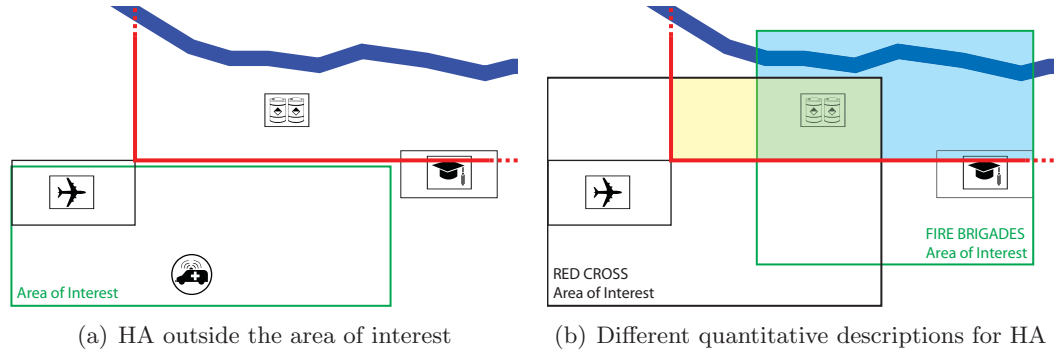


Figure 3.7: Drawbacks in the definition of areas of interest: (a) the quantification of a qualitative relation might be outside the area of interest, and (b) the quantification operation should return a different quantitative description for any defined area of interests.

The definition of an area of interest involves however two drawbacks, shown in Fig 3.7. The first shows up if an unknown entity is physically located outside the defined area of interest. In this case, also the quantification of the relations that

¹The *Open Geospatial Consortium* (<http://www.opengeospatial.org>) defines the expression *area of interest* as “A user defined area (represented by a bounding box, circle or polygon)”.

describe the unknown entity could result to be outside the area of interest, and the quantification would yield an empty object. As an example, if the area of interest for the ambulance driver is defined as the green box in Fig 3.7(a), the quantification of the relation $NE(O_{HA}^*, O_{AI})$ (constrained by the red line in the figure) intersected with the area of interest yields an empty object; thus, the quantitative description of the hazardous area HA is lost. The second disadvantage arises by considering that the spatial knowledge is shared with different actors, any of which has a different area of interest. Fire brigades could define a certain area of interest for their operations, while the red cross can set up a different area of interest (respectively the green and the black boxes in Fig. 3.7(b)). Thus, the quantification operation has to compute one quantitative description of the hazardous area for every defined area of interest (yellow area for the red cross and cyan area for the fire brigades). To summarize, allowing for the definition of areas of interest that constraint the result of the quantification would produce the counter-effect that the quantification output is not uniquely defined. A representation method for regions with infinite extent, that does not ask for a-priori definition of the areas of interest and that exploits functionalities already provided by GIS, is required instead.

3.3.3 Imprecise Description of Spatial Regions

Given an unknown entity represented by O_1^* and qualitatively described by an n -ary relation $R(O_1^*, O_2, \dots, O_n)$, the maximal quantification $A_R^+(O_2, \dots, O_n)$ results in the Euclidean region in which O_1^* has to be contained; conversely, $A_R^-(O_2, \dots, O_n)$ has to contain O_1^* . Hence, the objects resulting from the quantification of the relation can be used for defining the representation O_1^* .

The vector method is not suited for representing entities whose description is uncertain, as it is for the result of the quantification. However, considering that A_R^- represents the spatial region that surely is occupied by the real spatial entity, while the region contained between A_R^- and A_R^+ may contain the real entity, a three-valued logic approach is adopted to represent the quantification outcomes in the system. Furthermore, the condition $A_R^-(O_2, \dots, O_n) \subseteq A_R^+(O_2, \dots, O_n)$ is always satisfied, hence these spatial regions can be represented by the *egg-yolk* approach (Cohn & Gotts, 1996; Lehmann & Cohn, 1994).

The representation is based on a pair of spatial objects—that can be described as vector objects—that represent the minimal extension (the yolk) and the maximal extension (the egg) for an entity described with some kind of uncertainty. The object that represents the egg has to contain the object that describes the yolk. The boundary of the real entity has then to lie somewhere between the yolk and the egg: the entity boundary never overlaps the yolk and never overlaps the exterior of the egg. An egg-yolk object O^* is defined as a pair of vector objects (O^+, O^-) , with O^+ representing

the *maximal extension* of O^* and O^- representing its *minimal extension*. An example of an egg-yolk object is depicted in Fig. 3.8.

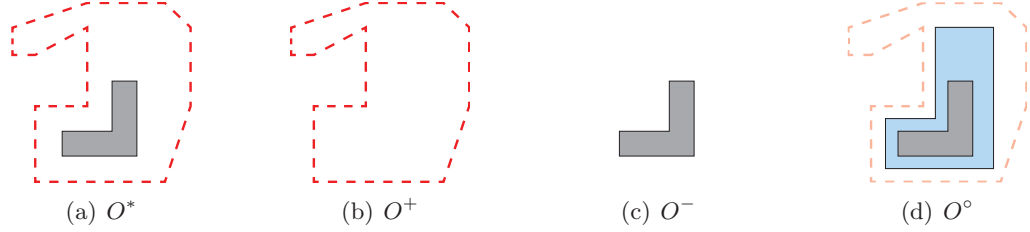


Figure 3.8: Egg-yolk object O^* . The objects O^+ and O^- represent respectively the maximal and minimal extension of the egg-yolk object. O^o is a precise instance of O^* .

If the description of the boundary of a spatial entity is acquired through traditional data collection methods, the description is assumed to be devoid of uncertainty. For instance, the description of the spatial region in Fig. 3.9(a) results in this case as the object depicted in Fig. 3.9(b); the description is equal to the true value. Representations such as the object in Fig. 3.9(c) are not admitted as representations of known entities. The resolution of the description depends from internal characteristics of the system (e.g., the number of digits after the decimal point that the system is able to represent). Thus, the known objects are precise and accurate descriptions of the spatial extent of known entities.

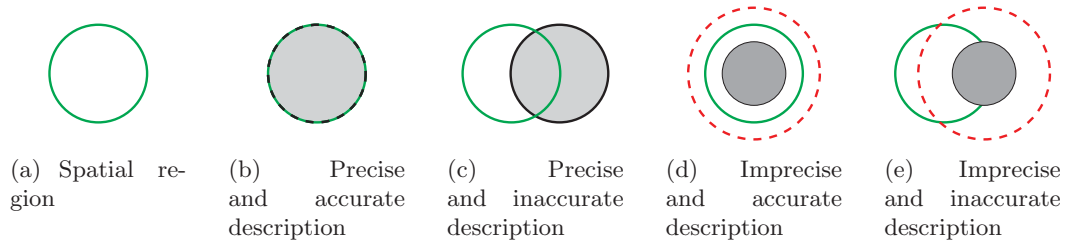


Figure 3.9: Precision and accuracy in the representation of a spatial region.

In contrast, an egg-yolk object provides only a range of possible values where the unknown entity can be located. However, the entity boundary is always contained within the egg and the yolk of the egg-yolk object (i.e., the representation in Fig. 3.9(d) obeys this constraint while the representation in Fig. 3.9(e) does not). Thus, the description is still accurate, but it is imprecise. As for the representation of known entities, the resolution depends on the internal characteristics of the system.

Being the precision the kind of uncertainty that differentiates the descriptions of known and unknown entities, egg-yolk objects are called *imprecise objects* in the re-

mainder of this thesis. As opposite, the spatial objects that describe known entities are called *precise objects*.

In principle, both O^+ and O^- can be multi-region objects; however, in the remainder of this text, for the most part objects whose egg and yolk consist of a single component without any holes will be used. It is further assumed that spatial entities in the reality always occupy a single region. This assumption does not imply a loss of generality since an entity that occupies more than one region in the space (multi-region) can be represented using distinct single-region objects. For instance, a fire that splits into two separate shares can be represented using an unknown object for each single share. Given an imprecise object O^* , for which O^+ and O^- are multi-region objects, from this assumption it directly follows that: (1) all the components of O^- are contained in only one component of O^+ , and (2) the description can be simplified considering as egg only the component that contains the yolks; hence the following axiom holds:

Axiom 3.3.1. *Given an imprecise object O^* , being O^+ a multi region object with components O_1^+, \dots, O_n^+ , and O^- a multi region object with components O_1^-, \dots, O_m^- , it exists k such that:*

$$(1) \forall i \forall j : 1 \leq i \leq n, 1 \leq j \leq m \Rightarrow \begin{cases} O_j^- \subseteq O_i^+ & \text{iff } i = k \\ O_j^- \cap O_i^+ = \emptyset & \text{iff } i \neq k \end{cases}$$

$$(2) O^* = (O^+, O^-) \equiv (O_k^+, O^-)$$

Examples of multi-region imprecise objects are depicted in Fig. 3.10. The imprecise object O_1^* (Fig. 3.10(a)) satisfies the first condition in Axiom 3.3.1 and can hence be simplified as in Fig. 3.10(b). The multi-region imprecise object O_2^* (Fig. 3.10(c)), instead, is not an admitted object in this work since it does not satisfy the first condition of the axiom.

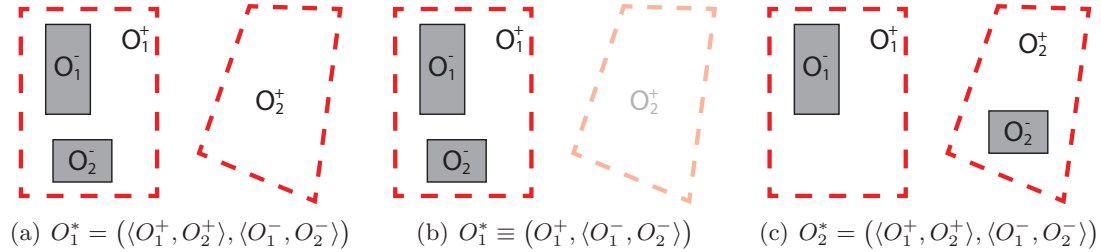


Figure 3.10: Multi-region imprecise objects.

If there is no knowledge about the maximal extension of an entity, a special representation for the complete plane \mathbb{R}^2 is used. Similarly, no knowledge about the minimal

extension is represented by a special representation for the empty set \emptyset . This allows for relaxing the constraint introduced by the egg-yolk representation that, generally, does not allow for empty yolk (Roy & Stell, 2001, p. 207). Furthermore, the case $O^+ = O^-$ (the egg and the yolk are coincident) corresponds to the egg-yolk definition of a precise region. In the remainder of this text, the following symbols will be used to differentiate the different cases: O^* is used to denote an imprecise object with non-empty yolk ($O^* = (O^+, O^-)$); O^+ denotes an imprecise object with empty yolk ($O^+ \equiv (O^+, \emptyset)$), and O defines a precise object (O is equivalent to the imprecise object defined by the pair (O, O)).

Finally, the simple regions that are contained in O^+ , and that contain O^- , are called *precise instances* (denoted by O°) of O^* (Fig. 3.8(d)).

Definition 10 (Precise instance of an imprecise object).

Given an imprecise object O^ , a precise object O° is called precise instance of O^* iff $O^- \subseteq O^\circ \subseteq O^+$.*

While O^- can be considered as the reliable part of an imprecise object, and can be interpreted as a precise object, the imprecision is expressed by the object O^+ . In order to refer to such an object, but interpreted as a precise object, the symbol O^\bullet will be used in this text; O^\bullet is called *precise interpretation* of O^+ . O^+ and O^\bullet have the same definition in geometric terms, but the former is interpreted as imprecise while the latter as precise. From Definition 10, the following lemma holds:

Lemma 3.3.1. *Let O^* be an imprecise object, and let O^\bullet be the precise interpretation of O^+ , the following conditions are satisfied for any precise instance O° of O^* :*

$$\forall O_2^- \subseteq O_2^\circ \subseteq O_2^+ \quad \begin{cases} \overline{X}(O_2^-) \leq \overline{X}(O_2^\circ) \leq \overline{X}(O_2^\bullet) \\ \overline{Y}(O_2^-) \leq \overline{Y}(O_2^\circ) \leq \overline{Y}(O_2^\bullet) \\ \underline{X}(O_2^-) \geq \underline{X}(O_2^\circ) \geq \underline{X}(O_2^\bullet) \\ \underline{Y}(O_2^-) \geq \underline{Y}(O_2^\circ) \geq \underline{Y}(O_2^\bullet) \end{cases} \quad (3.1)$$

Proof. The Lemma directly follows from the definitions of $\overline{X}(O_2)$, $\underline{X}(O_2)$, $\overline{Y}(O_2)$ and $\underline{Y}(O_2)$ (cf. Section 2.1.2.1). \square

The quantification of a qualitative relation $R(O, O_1, \dots, O_n)$ can be hence redefined as being an imprecise object $A_R^*(O_1, \dots, O_n)$, with its egg $A_R^+(O_1, \dots, O_n)$ and its yolk $A_R^-(O_1, \dots, O_n)$ defined as in Definition 9.

3.3.4 Quantification with Imprecise Reference Objects

The definition of the quantification based on clear geometric criteria provides a well-defined semantics to any relation. For instance, the maximal extension $A_N^+(O_2)$ for the relation $N(O_1^*, O_2)$ can be, in the case of precise object, defined as in Equation 3.2.

$$\begin{aligned} A_N^+(O_2) &= \bigcup_{\substack{O \subseteq \mathbb{R}^2 \\ N(O, O_2)}} O \\ &= \left\{ (p_x, p_y) \in \mathbb{R}^2 \mid \left(\forall (o_x, o_y) \in O_2 : p_y > o_y \right) \wedge \left(\exists (o_x, o_y) \in O_2 : p_x = o_x \right) \right\} \end{aligned} \quad (3.2)$$

Yet, if the reference object of the relation is imprecise, the definition can no longer be applied. Indeed, given for instance the relation $N(O_1^*, O_2^*)$ the definition of the quantification operation has to take into account the imprecision of the reference object by considering that the maximal quantification $A_N^+(O_2^*)$ is the union of the maximal quantifications $A_N^+(O_2^\circ)$ over all the precise instances of O_2^* (Equation 3.3). This result can be directly generalized for any n -ary relation R , as shown in Equation 3.4. Similarly, the minimal quantification of a relation having imprecise reference objects is defined as in Equation 3.5.

$$A_N^+(O_2^*) \triangleq \bigcup_{O_2^- \subseteq O_2^\circ \subseteq O_2^+} A_N^+(O_2^\circ) \quad (3.3)$$

$$A_R^+(O_2^*, \dots, O_n^*) \triangleq \bigcup_{\substack{O_2^- \subseteq O_2^\circ \subseteq O_2^+ \\ O_n^- \subseteq O_n^\circ \subseteq O_n^+}} A_R^+(O_2^\circ, \dots, O_n^\circ) \quad (3.4)$$

$$A_R^-(O_2^*, \dots, O_n^*) \triangleq \bigcap_{\substack{O_2^- \subseteq O_2^\circ \subseteq O_2^+ \\ O_n^- \subseteq O_n^\circ \subseteq O_n^+}} A_R^-(O_2^\circ, \dots, O_n^\circ) \quad (3.5)$$

However, the definitions are based on an infinite number of precise objects and, hence, they do not yield constructive procedures to build the quantification for the relation. What is required, instead, is to compute the quantification—or at least a very good approximation of it—by combining quantifications from precise cases in a suitable way, e.g., by interpreting the egg and yolk of the reference objects as precise objects. Furthermore, intermediate cases—in which, for instance, the yolk object is empty—need to be investigated as well.

3.3.5 Qualification

Opposite to the quantification operation, the *qualification* is the translation operation that transform quantitative information pieces into qualitative ones.

Definition 11 (Qualification between precise objects).

Given an n -ary calculus C , and a set of precise objects O_1, \dots, O_n , the qualification operation yields the relation, denoted as $R_C(O_1, \dots, O_n)$, that holds between O_1, \dots, O_n .

If the reference objects are precise, the qualification operation simply consists of verifying the constraint defined for every relation in the calculus. As an example, in order to qualify the CDC relation holding between two precise objects O_1 and O_2 , the definitions of the relations given in Equations 2.3-2.11 can be adopted. Nevertheless, those definitions fail when the relation holding between two imprecise objects has to be qualified. Most of the existing research focuses on the definition of *fuzzy relations* between imprecise objects (cf. Section 2.3.3), in other words new calculi are developed that define sets of relation holding between entities imprecisely described. For instance, Cohn & Gotts (1996) define a generalized topological calculus for egg-yolk objects—based on RCC-8—consisting of 46 base relations. Differently, Cicerone & Di Felice (2000) define a cardinal direction calculus, based on the CDC, for vague regions represented as broad boundary regions. By considering fuzzy relations between imprecise objects, the definition of the qualification operation given in Definition 11 does not undergo any change.

However, the entities considered in this work have well-defined boundaries in the reality, even though they can be imprecisely represented in the system as imprecise objects. Hence, the objective of the qualification is not the computation of fuzzy relations between imprecise objects, but rather the computation of the disjunctive *crisp relation*¹ that contains the relations holding between all the precise instances of the imprecise objects. For instance, considering the two imprecise objects in Fig. 3.11(a), there exist two precise instances O_1^o and O_2^o , with $O_1^- \subseteq O_1^o \subseteq O_1^+$ and $O_2^- \subseteq O_2^o \subseteq O_2^+$, such that either $DC(O_1^o, O_2^o)$ (Fig. 3.11(b)), or $EC(O_1^o, O_2^o)$ (Fig. 3.11(c)), or $PO(O_1^o, O_2^o)$ (Fig. 3.11(d)) hold. The qualification operation has to yield the disjunctive relation $R_{RCC}(O_1^*, O_2^*) = \{DC, EC, PO\}$. This result can be generalized for any n -ary qualitative calculus C , as in Definition 12.

Definition 12 (Qualification of the crisp relation between imprecise objects).

Given an n -ary calculus C , and a set of imprecise objects O_1^, \dots, O_n^* , the qualification operation yields the disjunctive relation:*

$$R_C(O_1^*, \dots, O_n^*) \triangleq \{R \in U_C \mid \forall i = 1 \dots n \exists O_i^o : O_i^- \subseteq O_i^o \subseteq O_i^+ \wedge R(O_1^o, \dots, O_n^o)\}$$

¹The term *crisp* is used here in the sense of *crisp set* as defined by Freksa (1994).

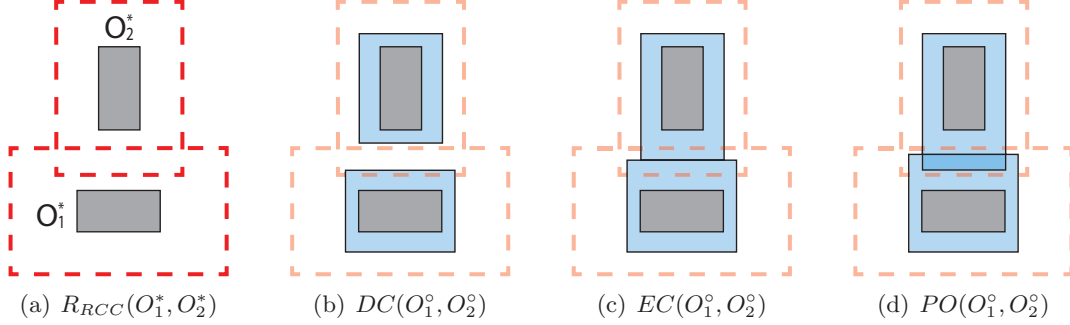


Figure 3.11: Qualification of relations between imprecise objects.

However, as for the quantification operation (cf. Sec. 3.3.4), this definition is based on an infinite number of precise objects and does not provide a computational procedure to retrieve the crisp disjunctive qualitative relation. The challenge is hence to derive the disjunctive relation from the relations holding between the eggs and the yolks of the different imprecise objects, interpreted as precise objects.

3.3.6 Reasoning

Given as input to the system a set of spatial objects O_1^*, \dots, O_n^{*1} and a set of qualitative relations describing how the different objects relate to each other, the reasoning capabilities, discussed in Section 2.2.2, can be exploited to boost the spatial knowledge available for the system.

As an example, the input sets depicted in Fig. 3.12(a) are considered. At the geometrical level, the input contains two precise objects (O_1, O_2) and two unknown objects (O_3^*, O_4^*) that can be initially described as imprecise objects having eggs set to \mathbb{R}^2 and yolks to \emptyset . Furthermore, three qualitative relations— $W(O_3^*, O_2)$, $N(O_3^*, O_4^*)$, and $E(O_4^*, O_1)$ —are given that relate O_1, O_2, O_3^* , and O_4^* . By assuming that a geometric description of O_3^* is required, a straightforward solution is to compute the quantification for the relations where O_3^* appears as primary object, that results in the region depicted in Fig. 3.12(b). In order to compute a quantitative representation for this region, algorithms to compute intersection and union operations of imprecise objects that potentially represent regions with infinite extent have to be developed. Those algorithms will be called *geometric reasoning* algorithms since they employ computational geometry techniques to refine the quantitative spatial knowledge of the system. However, this result only partially exploits the potential of the information available for

¹Note that any object can be either precise or imprecise. Even though the objects are all denoted as imprecise objects, a precise object can be represented by an imprecise object with coincident egg and yolk, as shown in Section 3.3.3.

the system. In fact, opportunely combining qualitative information (e.g., by adopting composition operations), new qualitative relations can be inferred or refined and the approximation of the object O_3^* can be refined as shown in Fig. 3.12(c).

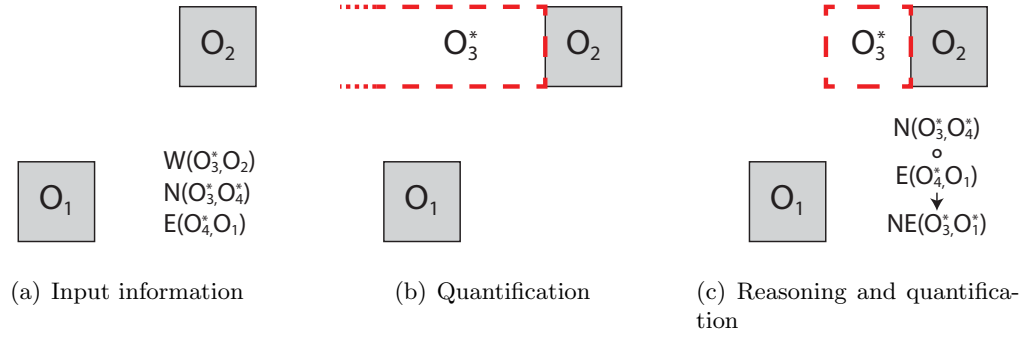


Figure 3.12: Reasoning with qualitative and geometric information.

Even though in this work the information is assumed to be consistent, consistency checking algorithms, such as Algebraic Closure (cf. Sec. 2.2.2.2), can be adopted to perform qualitative constraint propagation. Algebraic closure is suitable to reason with both binary (Mackworth, 1977) and ternary (Dylla & Moratz, 2004) calculi. However, those algorithms require a well-defined composition operation (e.g., in the ternary case the algebraic closure uses the composition $R(O_1, O_2, O_4) = R_1(O_1, O_2, O_3) \circ R_2(O_2, O_3, O_4)$). Such a composition is not always defined for all the calculi, i.e., the Visibility calculus defines a different composition operation, and, in principle, any calculus can define more than a composition operation. Hence, a generalized version of the algebraic closure algorithm is required that can deal with both binary and ternary calculi, each of which has different kinds of composition operations defined.

Furthermore, since more spatial aspects, and therefore different spatial calculi, are considered in this work, dependencies among different aspects should be taken into account in order to refine the system's knowledge. For instance, the relation $W(O_3^*, O_2)$ in Fig. 3.12(a) means also that $\{DC, EC\}(O_3^*, O_2)$. This qualitative relation can be combined with other topological information to propagate the acquired knowledge. In Section 2.2.4 it has been shown that different approaches exist to combine different qualitative spatial aspects, in particular combining the reasoning table or developing new calculi that take into account different characteristics. Those approaches, however, require a case-by-case investigation of the properties of the different aspects/calculi. Rather, geometrical properties of the calculi allow to automatically account for interdependencies among different spatial aspects into the system.

3.4 Summary

In this chapter, limits in the existing methods for acquiring updated geographical information after an extreme event have been highlighted in Section 3.1. However, recent studies in the field of VGI have shown how new instruments can be provided to volunteers in order to collect spatial data. These instruments are still subject to the main drawbacks in traditional collection methods, but they open the door to innovative ways for geographic data collection.

In this work, a framework to extract and interpret spatial information conveyed by reports from people in the field—such as the first responders to the extreme event—is proposed as a replacement for the traditional collecting methods that are not able to provide updated spatial information during the first hours—or even days—after the event. In particular, qualitative spatial calculi developed in the Qualitative Spatial Representation and Reasoning research field are proposed as means to interpret information contained in human communications.

The core problem analyzed in this work is the one related to the integration of qualitative spatial information with quantitative information stored in a geometric way into existing Geographic Information Systems. Challenges related to this task can be broadly grouped into three main operations: quantification, qualification, and reasoning. The quantification operation allows for transforming qualitative information into quantitative information by eventually dealing with either or both regions having infinite extent and imprecise descriptions; quantification will be presented in Chapter 4. Instead, Chapter 5 will present the operation of qualification, that computes the disjunctive qualitative spatial relation holding between a given set of precise and/or imprecise objects. Finally, in Chapter 6 a hybrid qualitative-quantitative reasoning system will be shown that boosts the system’s spatial knowledge by combining qualitative spatial reasoning capabilities with computational geometry algorithms.

Chapter 4

Quantification of Qualitative Spatial Information

In the previous chapter a system for the integration of geographic information has been proposed. The focus of this work is on the development of the *geographic information integration layer* of the system. Given a set of known and unknown spatial entities and a set of information that partially describes the entities in qualitative and quantitative terms, the layer's purpose is the integration of the mixed spatial information.

A first design of the layer is done in Section 4.1 by considering problems of spatial information translation¹. Afterwards, the focus of this chapter is shifted to the quantification component development: Section 4.2 analyzes the quantification problem and describes which are the challenges in the development of computational procedures for the quantification of qualitative relations. Section 4.3 describes the representation approach adopted in this work to model spatial regions having infinite extent. The properties of the spatial regions that the integration system is able to represent will be then summarized in Section 4.4. The *Spatial-region object* is defined to represent a spatial region whose extent is either finite or infinite, that may have several disconnected components and holes, and whose description is either precise or imprecise. A general algorithm for the quantification of a single qualitative relation is proposed in Section 4.5, while Section 4.6 focuses on the specific approaches to quantify topological, cardinal direction and visibility relations. Finally, Section 4.7 analyzes the computational complexity of the proposed algorithms.

¹As defined in Section 2.2.5, the term *translation* is used in this text to denote those operations that perform a transformation of spatial information from one representation approach—qualitative or quantitative—to the other.

4.1 A System for Spatial Information Translation

A system for the integration of geographic information has been proposed in Section 3.2.3 that is composed of a *Storage Layer* for the permanent storage of spatial information, and a *Geographic Information Integration Layer* for the integration of qualitative and quantitative information. The extensions of both known and unknown entities (the spatial entities are either precisely or imprecisely represented) are described within the storage layer by means of imprecise objects (cf. Section 3.3.3).

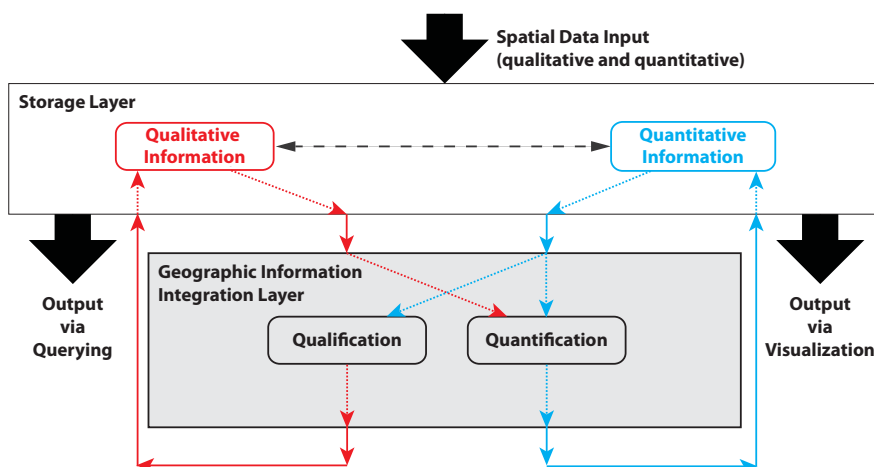


Figure 4.1: A system to translate geographic information¹.

At first, the geographic information integration layer is designed as a layer that performs translation operations among qualitative and quantitative spatial information². Two distinct components build the layer: the *quantification component* and the *qualification component*, as Fig. 4.1 shows. The input to the former component is composed of an n -ary qualitative spatial relation, and a set of $n - 1$ spatial objects that are the reference objects of the given relation; the component produces as output a geometric description of the relation as an imprecise object (cf. Definition 9 p. 58). The qualification component, instead, gets as input an n -ary calculus C defined over a domain of connected regions in \mathbb{R}^2 , and a set of n spatial objects; it yields the n -ary qualitative spatial relation defined in C that holds between the set of objects (cf. Definition 12 p. 65)³.

¹The red lines in the figure represent the flow of qualitative information: solid lines show the stream of information between the different system's layers, while dotted ones depict how the information flows within any single layer. In the same way, the cyan lines represent the stream of quantitative information. The roles of the different layers have been described in Fig. 3.5.

²The system proposed in this chapter will be extended in Chapter 6.

³The qualification component will be discussed in Chapter 5.

For both translation procedures, approaches for dealing with precise objects need to be extended in order to adequately deal with imprecise objects.

4.2 Quantification Challenges

The *quantification* operation has been defined by Wolter & Wallgrün (2012) as the process of computing one exemplary quantitative scene description for a given qualitative scene description. For instance, given the qualitative relations $E(O_5^*, O_1^*)$, $E(O_4^*, O_2)$, $N(O_4^*, O_3)$ and $N(O_5^*, O_4^*)$ holding between the two unknown entities O_4^* , O_5^* and the three known entities O_1 , O_2 and O_3 , the quantification aims at finding exemplary precise descriptions for O_4^* and O_5^* that satisfy all given relations. Both scenes in Fig. 4.2(a) and Fig. 4.2(b) show possible results of the quantification process¹. A similar approach has been used by Steinhauer (2008) that partitions the space into regular cells and assumes that the entities have fixed dimension equals to the grid cells; hence, any entity can be located into exactly one grid cell.

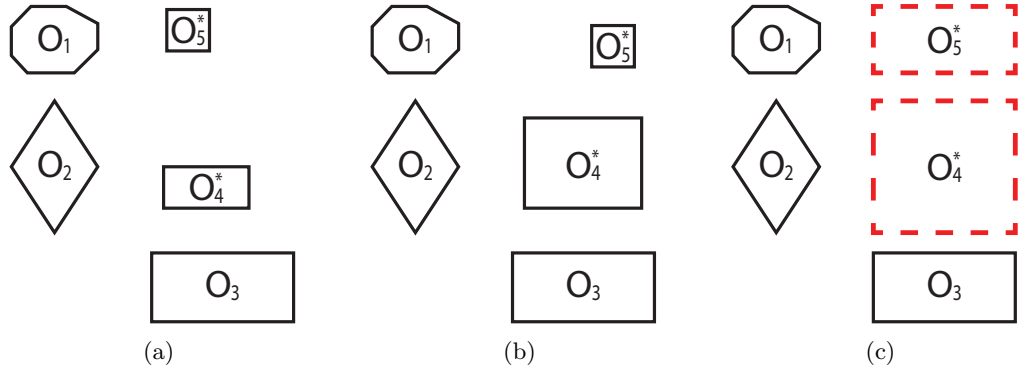


Figure 4.2: Quantification of CDC relations: Exemplary quantitative scene descriptions (a)-(b) and maximal quantification (c).

In contrast to Steinhauer (2008); Wolter & Wallgrün (2012), the target of the quantification in this work is not the computation of one exemplary precise quantitative scene description. Rather, it is required to compute two objects to describe any unknown entity: one representing the region in \mathbb{R}^2 where the entity has to be contained in order to satisfy all relations in the set, and one representing the region that has to be contained into the spatial region occupied by the unknown entity. The result of the quantification for the relations above is depicted in Fig. 4.2(c). In particular, this chapter focuses on the quantification of a single qualitative spatial relation, while the quantification

¹The quantitative interpretation is based on the semantics of the relations as defined in the CDC calculus (cf. Section 2.2.3.2).

of a qualitative scene description is analyzed in Chapter 6. In the remainder of this section, the quantification of a single qualitative relation is formally defined and the challenges in the development of computational procedures for the quantification of a single qualitative relation will be shown.

Let O_2^*, \dots, O_n^* be a set of spatial objects each of which can be either precise or imprecise, and let R be a qualitative relation defined in an n -ary calculus C defined over a domain of connected regions in \mathbb{R}^2 , the quantification operation consists of identifying two regions in \mathbb{R}^2 :

MAXIMAL QUANTIFICATION. It is the region in \mathbb{R}^2 that *contains* all the precise objects $O_i \subseteq \mathbb{R}^2$ that satisfy $R(O_i, O_2^*, \dots, O_n^*)$. It is denoted as $A_R^+(O_2^*, \dots, O_n^*)$.

MINIMAL QUANTIFICATION. It is the region in \mathbb{R}^2 that *is contained in* all the precise objects $O_i \subseteq \mathbb{R}^2$ that satisfy $R(O_i, O_2^*, \dots, O_n^*)$. It is denoted as $A_R^-(O_2^*, \dots, O_n^*)$.

To give an example, Fig. 4.3(a) depicts some exemplary precise objects that satisfy the CDC relation N with respect to the precise object O_2 . In this case, the maximal quantification is equivalent to the acceptance area defined for the relation N in the CDC calculus, corresponding to the shaded area in Fig. 4.3(b).

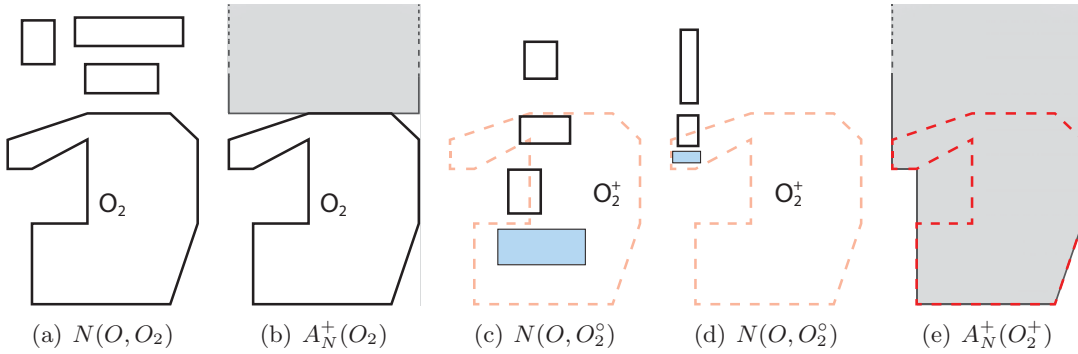


Figure 4.3: Quantification A_N^+ if the reference object is precise or imprecise.

In contrast, if the reference object is imprecise, all its precise instances have to be considered while computing the quantification. Indeed, an imprecise object represents an entity for which a precise description is not known; hence, any precise instance potentially represents the real extent of the spatial entity. Thus, the maximal quantification of a relation having an imprecise reference object O_2^* has been defined in Section 3.3.4 as the union of the maximal quantifications computed over all the precise instances of O_2^* . Similarly, the minimal quantification is computed by means of intersection operations.

Fig. 4.3(c) and Fig. 4.3(d) show exemplary precise objects that satisfy the relation N with respect to two different precise instances of the imprecise object O_2^+ . The

maximal quantification $A_N^+(O_2^+)$ is hence the shaded region shown in Fig. 4.3(e) that considers all possible precise instances of O_2^+ . However, the quantification definition is based on an infinite number of precise instances, thus it does not yield a constructive procedure for building $A_N^+(O_2^+)$ and $A_N^-(O_2^+)$. The trivial approach of adopting multiple sorting for computing the quantification, as proposed by Wolter & Wallgrün (2012), would yield an object that includes also the region corresponding to $MBR(O_2^+)$. Hence, more sophisticated procedures have to be developed for the quantification of qualitative relations in case the reference objects are imprecise.

In the following sections, quantification procedures are proposed for cardinal direction, visibility, and topological relations. Beforehand, the definition of a representation approach for dealing with regions having infinite extent, as the maximal quantifications shown in Fig. 4.3(b)-(e), is required.

4.3 Representation of Regions with Infinite Extent

The focus of this section is on the definition of the approach adopted in this work for dealing with regions having infinite extent, whose representations are called *Infinite-Region Objects*. Indeed, as shown in the previous section, the quantification operation often yields regions that extend to infinity. For instance, the relation $N(O_1, O_2)$ together with the object O_2 depicted in Fig. 4.4(a) implies that O_1 has to lie somewhere in the shaded region which reaches out to infinity. Vector representations as commonly implemented in GIS (Herring, 2001) do not support the representation and manipulation of such kinds of infinite regions. Such regions can be represented using a half-plane approach (Rigaux & Scholl, 1995; Rigaux *et al.*, 2002) in which objects are represented by means of Boolean operations between half-planes in \mathbb{R}^2 . Frank *et al.* (1996); Haunold *et al.* (1997) propose efficient data structures for the representation of half-planes in GIS. However, it turns out that it is sufficient to make a small modification to the vector representation adopting the idea of half-plane representation. The advantage of this approach is that existing procedures—provided by most of the existing GIS—to perform geometric procedures with multi-region objects can still be applied.

A *simple infinite-region object* IR is defined as a triple $(\Lambda, \vec{\lambda}_1, \vec{\lambda}_2)$, in which the polyline $\Lambda = \langle p_1, \dots, p_n \rangle$ represents the finite boundary of IR , while the two rays $\vec{\lambda}_1 = [p_1, q)$ and $\vec{\lambda}_2 = [p_n, r)$ define the boundaries of IR that extend to infinity. The starting point of Λ corresponds with the starting point of $\vec{\lambda}_1$. In the same way, the last point in Λ corresponds to the starting point of $\vec{\lambda}_2$.

An infinite-region object introduces a partition of the space into two infinite regions. The actually represented region is the intersection of the half-plane right of $\vec{\lambda}_1$, the half-plane left of $\vec{\lambda}_2$ and what can intuitively be seen as the area left of the polyline Λ . Fig. 4.4(d) shows the infinite-region object representing the acceptance area for

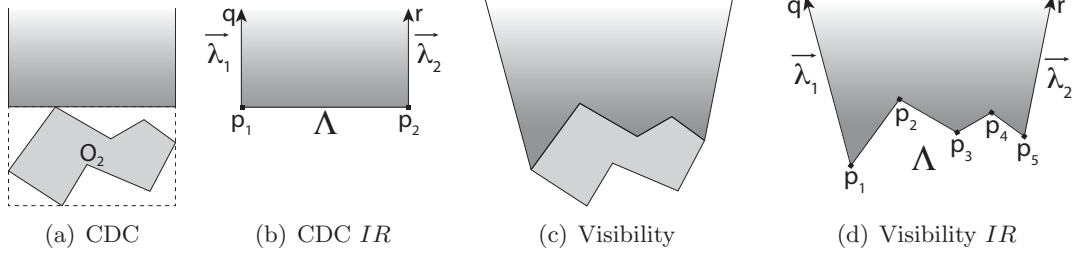


Figure 4.4: Infinite acceptance areas and the corresponding infinite-region objects.

the visibility relation *Occluded* as depicted in Fig. 4.4(c). It consists of the three components $\Lambda = \langle p_1, p_2, p_3, p_4, p_5 \rangle$, $\vec{\lambda}_1 = [p_1, q]$, and $\vec{\lambda}_2 = [p_5, r]$.

Algorithms to perform the computation of intersection, union, and difference of two infinite-region objects as well as of a polygon and an infinite-region object are analyzed in Appendix A. Let IR_1 and IR_2 be two infinite-region objects defined respectively by n and m vertices, the computational time complexity of the algorithms is $\mathcal{O}(nm)$, that is equivalent to the time complexity of the same operations between spatial objects with finite extent (Margalit & Knott, 1989). Intersection, union, and difference operations potentially yield regions having more components and holes as for the vector representation. Hence, Appendix A describes how two represent complex regions with infinite extent—that may have more components and holes—by means of infinite-region objects; also, it is shown how regions with finite extent can be represented by infinite-region objects.

4.4 Spatial-Region Objects

Different properties of spatial regions have so far been introduced and discussed in this text: single or multi-component, precisely or imprecisely described, having either finite or infinite extent, and with or without holes. Hence, 16 types of regions can exist, and different approaches to represent spatial regions with specific properties have been proposed (i.e., multi-polygons, infinite-region objects, and imprecise objects). However, a common representation method—called *spatial-region object*—that carries on all the necessary properties can be employed. Indeed, a single component spatial region can be represented as a multi-component one having only one component. Similarly, both precise and imprecise descriptions of spatial regions can be represented by egg-yolk objects, while regions with finite extent can be easily represented using infinite-region objects. Hence, the symbols O_1^*, \dots, O_n^* are used to denote *spatial-region objects*¹ that

¹The symbols O_1^*, \dots, O_n^* have been used in the previous chapter to denote unknown/imprecise objects. However, the usage of the same symbols to refer to spatial-region objects does not introduce

are imprecise objects, whose eggs and yolks are multi-infinite-region objects eventually with holes. The conventional notations defined in Section 3.3.3 are still valid. For the sake of simplicity, from now on—where it is not differently specified—it is assumed that spatial-region objects are single-finite-region objects without holes.

4.5 The Quantification Component

The quantification component in Fig 4.1 computes a geometric interpretation of a single n -ary relation tuple $R(O_1^*, \dots, O_n^*)$ in which O_1^* is the primary object and O_2^*, \dots, O_n^* are the reference objects. The computation is based on the geometric semantics of the relation and has to take into account the imprecise descriptions of the involved reference regions. The result is a new imprecise object that represents the geometric information about O_1^* as implied by this particular relation.

4.5.1 Quantification Algorithm

Given an n -ary relation R belonging to a calculus C defined over a domain of connected regions in \mathbb{R}^2 and that holds between O_1^* and the spatial-region objects O_2^*, \dots, O_n^* , a function $Quantify(R, C, O_2^*, \dots, O_n^*)$ is defined to compute a geometric description of O_1^* . The semantics of R determines whether the quantification process influences either or both the maximal and the minimal extensions of O_1^* . The topological relation $DC(O_1^*, O_2)$, for instance, between an unknown object O_1^* and a precise object O_2 , defines a constraint for O_1^+ as the unknown entity can be anywhere except where it would overlap with O_2 . It does, however, not provide any information about the minimal extension O_1^- . Conversely, for $NTPPI(O_1^*, O_2)$ the opposite holds. Projective and metric qualitative relations typically only provide information about the maximal extension of an unknown entity.

The general procedure to compute $Quantify(R, C, O_2^*, \dots, O_n^*)$ is shown in Algorithm 3. A calculus-dependent function $Quantify_C(br, O_2^*, \dots, O_n^*)$ is used to retrieve a geometric description for the quantification of a base relation br contained in R . Since R can be a disjunction of base relations, the quantified object for R is constructed combining the quantifications for each base relation $br \in R$.

The imprecision of the reference objects has to be taken into account when a base relation is quantified by $Quantify_C$. The $Quantify_C$ functions for topology, cardinal direction and visibility relations will be described in the next section.

any ambiguity in the discussion. Indeed, only the extent that the eggs and the yolks can have is redefined with respect to the definition given in Section 3.3.3.

Algorithm 3 *Quantify*($R, C, O_2^*, \dots, O_n^*$)

```

 $O_1^* \leftarrow (\emptyset, \mathbb{R}^2)$ 
for  $br \in R$  do
   $Q^* \leftarrow \text{Quantify}_C(br, O_2^*, \dots, O_n^*)$ 
   $O_1^+ \leftarrow O_1^+ \cup Q^+$ 
   $O_1^- \leftarrow O_1^- \cap Q^-$ 
end for
return  $O_1^*$ 

```

4.6 Quantification of Qualitative Spatial Relations

If the reference objects of a given qualitative relation are precise, the *Quantify_C* function can be implemented by giving a geometric interpretation of the relation semantics as defined in the calculus (e.g., Clementini & Billen, 2006; Skiadopoulos & Koubarakis, 2004). For instance the maximal quantification of the relation $N(O_1^*, O_2)$ corresponds to the geometric description of the acceptance area of the relation as defined in the CDC calculus. However, for an imprecise reference object O_2^* , this definition has to be modified to take into account the imprecision by saying that the quantification $A_N^+(O_2^*)$ is the union of the quantifications $A_N^+(O_2^\circ)$ over all precise instances of O_2^* . However, as has already been shown this definition is based on an infinite number of precise objects and does not yield a constructive procedure to build the quantification of the relation. In this section, a solution to compute the quantification by combining quantifications from precise cases in a suitable way will be presented. The final step of coming up with a constructive definition so far had to be conducted on a case-by-case analysis in which the relations' defining conditions have been compared to the properties that can be derived for the respective quantifications $A_N^+(O_2^-)$, $A_N^+(O_2^\bullet)$, etc. under the side condition that $O_2^- \subseteq O_2^\bullet$. As an intermediate step the quantification $A_N^+(O_2^+)$ has been defined which assumes that O_2^+ describes the maximal extension of the reference object but also assumes that the yolk is empty. The result is always a formula in which different quantifications are combined via intersection and union.

4.6.1 *Quantify_C* for Cardinal Direction Relations

The Cardinal Direction Calculus (Goyal & Egenhofer, in press) defines 218 base relations (cf. Section 2.2.3.2); among them, nine relations are single-tile while the others are defined as multi-tile relations. At first, it trivially results that the minimal quantification of a CDC relation is always an empty object:

$$A_R^-(O_2^*) = \bigcap_{O \subseteq \mathbb{R}^2 \mid R(O, O_2^*)} O = \emptyset \quad \text{with } R \in \mathcal{R}_{CDC}$$

This result is directly proven by considering the precise objects in Fig. 4.5. Both O'_1 and O''_1 satisfy the cardinal direction relation N with respect to O_2 . Furthermore, $O'_1 \cap O''_1 = \emptyset$ from which it directly results $A_N^-(O_2) = \emptyset$. The same result is valid for all the relations in the CDC calculus. Thus, the quantification of a CDC relation always yields a spatial-region object having empty yolk.

Moreover, the maximal quantification A_R^+ of a single-tile relation over a precise object corresponds to the acceptance area defined for that relation in the model. For a multi-tile relation, instead, the maximal quantification is given by the union of the quantifications of any single-tile relation that composes it. Hence, only the quantifications of the base relations $R \in \mathcal{R}_{CDC}^{ST}$ require to be analyzed.

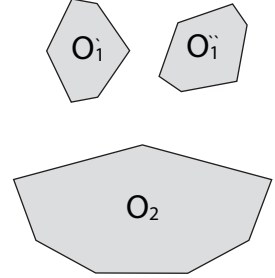


Figure 4.5: $A_R^-(O_2)$.

Algorithm 4 *Quantify_C*(R, O_2^*) – Cardinal Direction Calculus

```

 $A_R^* \leftarrow (\emptyset, \emptyset)$ 
for  $R^{ST} \in \Gamma(R)$  do
     $A_R^+ \leftarrow A_R^+ \cup A_{R^{ST}}^+(O_2^*)$ 
end for
return  $A_R^*$ 

```

The *Quantify_C* procedure to quantify any base cardinal direction relation is shown in Algorithm 4¹. The computation of $A_R^+(O_2^*)$, with $R \in \mathcal{R}_{CDC}^{ST}$, under the assumption that O_2^* is a precise object will be discussed first. Afterwards, the results will be exploited for the quantification of CDC single-tile relations if O_2^* is an imprecise object with empty yolk. Finally, the general case in which O_2^* is an imprecise object with non-empty egg and yolk is discussed.

4.6.1.1 Precise Reference Object

If the reference object of a cardinal direction single-tile relation is precise, the relation's maximal quantification can be trivially defined both in formal terms and by means of infinite-region objects. At first, from Definition 9, the maximal quantification is defined for CDC single-tile relations as:

$$A_R^+(O_2) = \bigcup_{O \subseteq \mathbb{R}^2 \mid R(O, O_2)} O \text{ with } R \in \mathcal{R}_{CDC}^{ST} \quad (4.1)$$

¹Let R be a single or multi-tile base relation, the function $\Gamma(R)$ yields the tiles that compose R (cf. Section 2.2.3.2).

The definition is equivalent to the acceptance areas defined in Fig. 2.10 that are formally defined, from Eq. 4.1 and Eqs. 2.3-2.11, as:

$$A_B^+(O_2) = \{(p_x, p_y) \in \mathbb{R}^2 \mid \underline{X}(O_2) \leq p_x \leq \overline{X}(O_2) \wedge \underline{Y}(O_2) \leq p_y \leq \overline{Y}(O_2)\} \quad (4.2)$$

$$A_N^+(O_2) = \{(p_x, p_y) \in \mathbb{R}^2 \mid \underline{X}(O_2) \leq p_x \leq \overline{X}(O_2) \wedge p_y \geq \overline{Y}(O_2)\} \quad (4.3)$$

$$A_E^+(O_2) = \{(p_x, p_y) \in \mathbb{R}^2 \mid p_x \geq \overline{X}(O_2) \wedge \underline{Y}(O_2) \leq p_y \leq \overline{Y}(O_2)\} \quad (4.4)$$

$$A_S^+(O_2) = \{(p_x, p_y) \in \mathbb{R}^2 \mid \underline{X}(O_2) \leq p_x \leq \overline{X}(O_2) \wedge p_y \leq \underline{Y}(O_2)\} \quad (4.5)$$

$$A_W^+(O_2) = \{(p_x, p_y) \in \mathbb{R}^2 \mid p_x \leq \underline{X}(O_2) \wedge \underline{Y}(O_2) \leq p_y \leq \overline{Y}(O_2)\} \quad (4.6)$$

$$A_{NE}^+(O_2) = \{(p_x, p_y) \in \mathbb{R}^2 \mid p_x \geq \overline{X}(O_2) \wedge p_y \geq \overline{Y}(O_2)\} \quad (4.7)$$

$$A_{SE}^+(O_2) = \{(p_x, p_y) \in \mathbb{R}^2 \mid p_x \geq \overline{X}(O_2) \wedge p_y \leq \underline{Y}(O_2)\} \quad (4.8)$$

$$A_{SW}^+(O_2) = \{(p_x, p_y) \in \mathbb{R}^2 \mid p_x \leq \underline{X}(O_2) \wedge p_y \leq \underline{Y}(O_2)\} \quad (4.9)$$

$$A_{NW}^+(O_2) = \{(p_x, p_y) \in \mathbb{R}^2 \mid p_x \leq \underline{X}(O_2) \wedge p_y \geq \overline{Y}(O_2)\} \quad (4.10)$$

Alternatively, the quantifications $A_R^+(O_2)$, for all $R \in \mathcal{R}_{CDC}^{ST}$, can be described by means of infinite-region objects that are defined as in Fig. 4.6 and Eq. 4.11.

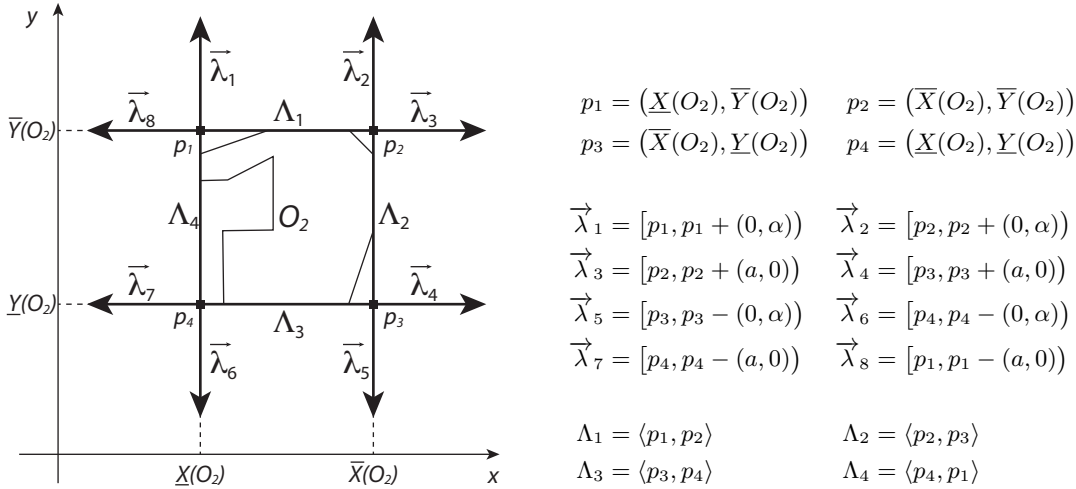


Figure 4.6: IR definition of the CDC quantification over a precise reference object.

$$\begin{aligned}
 A_{NW}^+(O_2) &= (p_1, \vec{\lambda}_8, \vec{\lambda}_1) & A_N^+(O_2) &= (\Lambda_1, \vec{\lambda}_1, \vec{\lambda}_2) & A_{NE}^+(O_2) &= (p_2, \vec{\lambda}_2, \vec{\lambda}_3) \\
 A_W^+(O_2) &= (\Lambda_4, \vec{\lambda}_7, \vec{\lambda}_8) & A_B^+(O_2) &= \langle p_1, p_2, p_3, p_4, p_1 \rangle & A_E^+(O_2) &= (\Lambda_2, \vec{\lambda}_3, \vec{\lambda}_4) \\
 & & &= MBR(O_2) & & \\
 A_{SW}^+(O_2) &= (p_4, \vec{\lambda}_6, \vec{\lambda}_7) & A_S^+(O_2) &= (\Lambda_3, \vec{\lambda}_5, \vec{\lambda}_6) & A_{SE}^+(O_2) &= (p_3, \vec{\lambda}_4, \vec{\lambda}_5)
 \end{aligned} \quad (4.11)$$

4.6.1.2 Imprecise Reference Object with Empty Yolk

The definitions of the maximal quantification require to be modified in the case if the reference object is imprecise. In this section it is considered only the intermediate case in which the imprecise reference object has empty yolk ($O_2^- = \emptyset$). As shown above, the quantification of a CDC relation R_1 always yields an object with empty yolk; thus, if the resulting object is the reference of a second relation R_2 , the quantification of R_2 has to deal with an object with empty yolk.

Let R be a CDC single-tile relation for which the reference object O_2^+ has empty yolk, the conditions expressed in Equation 3.4 can be relaxed as:

$$A_R^+(O_2^+) = \bigcup_{O_2^\circ \subseteq O_2^+} A_R^+(O_2^\circ) \quad \text{with } R \in \mathcal{R}_{CDC}^{ST} \quad (4.12)$$

Some auxiliary functions for describing the maximal quantifications of direction relations over a line segment by means of infinite-region objects are defined. Union operations between the resulting objects are then performed to compute $A_R^+(O_2^+)$ and it will be proven that the procedures yield objects equivalent to Equation 4.12.

At first, a function $BL(O)$ is defined that given a precise object O yields the list of its defining line segments.

Definition 13 (Boundary Line Segments). *Let O be a precise object defined by the list of points $\langle p_1, \dots, p_n, p_1 \rangle$, the function $BL(O)$ is defined as:*

$$BL(O) = \{[p_i, p_{i+1}] \mid i = 1 \dots n\}$$

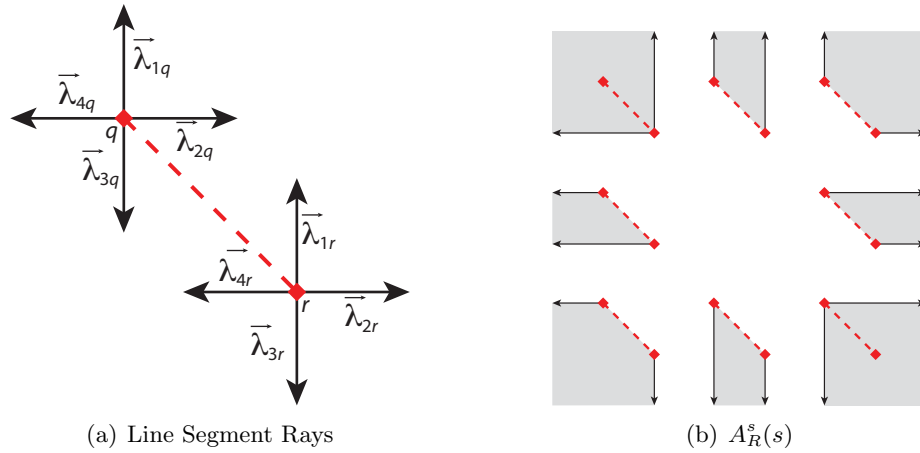


Figure 4.7: Line segment rays and quantifications over a line segment.

For any line segment, eight different rays—called *Line Segment Rays*—are defined that irradiate from either the segment's starting point or its ending point, and are parallel either to the x -axis or to the y -axis; Fig. 4.7(a) shows the line segment rays defined over an exemplary line segment.

Definition 14 (Line Segment Rays). *Let $s = [q, r]$ be a line segment, and let α be an arbitrary positive value, eight different line segment rays are defined on q and r :*

$$\begin{aligned} \vec{\lambda}_{1q} &= [q, q + (0, \alpha)] & \vec{\lambda}_{2q} &= [q, q + (\alpha, 0)] & \vec{\lambda}_{3q} &= [q, q + (0, -\alpha)] & \vec{\lambda}_{4q} &= [q, q + (-\alpha, 0)] \\ \vec{\lambda}_{1r} &= [r, r + (0, \alpha)] & \vec{\lambda}_{2r} &= [r, r + (\alpha, 0)] & \vec{\lambda}_{3r} &= [r, r + (0, -\alpha)] & \vec{\lambda}_{4r} &= [r, r + (-\alpha, 0)] \end{aligned}$$

A set of functions $A_R^s(s)$, with $R \in \mathcal{R}_{CDC}^{ST} \setminus B$, are defined that get as input a line segment s and yield the CDC *line segment quantifications* over s . The functions return the objects composed by all the points in \mathbb{R}^2 for which the following condition is satisfied: $A_R^s = \{p \in \mathbb{R}^2 \mid \exists q \in s : R(p, q)\}$; Fig 4.7(b) shows the line segment quantifications if the line segment in Fig 4.7(a) is considered as input for the functions.

Lemma 4.6.1. *Let $s = [p_1, p_2]$ be a line segment with $p_1 \neq p_2$, the functions $A_R^s(s)$, with $R \in \{N, E, S, W, NE, SE, SW, NW\}$, return the objects:*

$$\begin{aligned} A_N^s(s) &= (\langle q, r \rangle, \vec{\lambda}_{1q}, \vec{\lambda}_{1r}) \text{ with } \begin{cases} q = p_1, r = p_2 & \text{iff } p_{1x} < p_{2x} \\ q = p_1, r = p_1 & \text{iff } p_{1x} = p_{2x} \\ q = p_2, r = p_1 & \text{otherwise} \end{cases} \\ A_E^s(s) &= (\langle q, r \rangle, \vec{\lambda}_{2q}, \vec{\lambda}_{2r}) \text{ with } \begin{cases} q = p_1, r = p_2 & \text{iff } p_{1y} > p_{2y} \\ q = p_1, r = p_1 & \text{iff } p_{1y} = p_{2y} \\ q = p_2, r = p_1 & \text{otherwise} \end{cases} \\ A_S^s(s) &= (\langle q, r \rangle, \vec{\lambda}_{3q}, \vec{\lambda}_{3r}) \text{ with } \begin{cases} q = p_1, r = p_2 & \text{iff } p_{1x} > p_{2x} \\ q = p_1, r = p_1 & \text{iff } p_{1x} = p_{2x} \\ q = p_2, r = p_1 & \text{otherwise} \end{cases} \\ A_W^s(s) &= (\langle q, r \rangle, \vec{\lambda}_{4q}, \vec{\lambda}_{4r}) \text{ with } \begin{cases} q = p_1, r = p_2 & \text{iff } p_{1y} < p_{2y} \\ q = p_1, r = p_1 & \text{iff } p_{1y} = p_{2y} \\ q = p_2, r = p_1 & \text{otherwise} \end{cases} \\ A_{NE}^s(s) &= (\langle q, r \rangle, \vec{\lambda}_{1q}, \vec{\lambda}_{2r}) \text{ with } \begin{cases} q = p_1, r = p_2 & \text{iff } (p_{1x} < p_{2x} \wedge p_{1y} \geq p_{2y}) \\ & \vee (p_{1x} = p_{2x} \wedge p_{1y} > p_{2y}) \\ q = p_1, r = p_1 & \text{iff } p_{1x} < p_{2x} \wedge p_{1y} < p_{2y} \\ q = p_2, r = p_2 & \text{iff } p_{1x} > p_{2x} \wedge p_{1y} > p_{2y} \\ q = p_2, r = p_1 & \text{otherwise} \end{cases} \\ A_{SE}^s(s) &= (\langle q, r \rangle, \vec{\lambda}_{2q}, \vec{\lambda}_{3r}) \text{ with } \begin{cases} q = p_1, r = p_2 & \text{iff } (p_{1x} > p_{2x} \wedge p_{1y} \geq p_{2y}) \\ & \vee (p_{1x} = p_{2x} \wedge p_{1y} > p_{2y}) \\ q = p_1, r = p_1 & \text{iff } p_{1x} < p_{2x} \wedge p_{1y} > p_{2y} \\ q = p_2, r = p_2 & \text{iff } p_{1x} > p_{2x} \wedge p_{1y} < p_{2y} \\ q = p_2, r = p_1 & \text{otherwise} \end{cases} \end{aligned}$$

$$\begin{aligned}
A_{SW}^s(s) &= (\langle q, r \rangle, \vec{\lambda}_{3q}, \vec{\lambda}_{4r}) \text{ with } \begin{cases} q = p_1, r = p_2 & \text{iff } (p_{1x} > p_{2x} \wedge p_{1y} \leq p_{2y}) \\ & \vee (p_{1x} = p_{2x} \wedge p_{1y} < p_{2y}) \\ q = p_1, r = p_1 & \text{iff } p_{1x} > p_{2x} \wedge p_{1y} > p_{2y} \\ q = p_2, r = p_2 & \text{iff } p_{1x} < p_{2x} \wedge p_{1y} < p_{2y} \\ q = p_2, r = p_1 & \text{otherwise} \end{cases} \\
A_{NW}^s(s) &= (\langle q, r \rangle, \vec{\lambda}_{4q}, \vec{\lambda}_{1r}) \text{ with } \begin{cases} q = p_1, r = p_2 & \text{iff } (p_{1x} < p_{2x} \wedge p_{1y} \leq p_{2y}) \\ & \vee (p_{1x} = p_{2x} \wedge p_{1y} > p_{2y}) \\ q = p_1, r = p_1 & \text{iff } p_{1x} > p_{2x} \wedge p_{1y} < p_{2y} \\ q = p_2, r = p_2 & \text{iff } p_{1x} < p_{2x} \wedge p_{1y} > p_{2y} \\ q = p_2, r = p_1 & \text{otherwise} \end{cases}
\end{aligned}$$

Finally, it can be trivially proven that a precise object O is always contained in the union of the line segment quantifications computed over all segments that define O .

Lemma 4.6.2. *Let O_2 be a precise object, $\forall R \in \mathcal{R}_{CDC}^{ST} \setminus B$, $O_2 \subset \bigcup_{s \in BL(O_2)} A_R^s(s)$.*

The boundary line segments definition, the line segment quantifications, and the result of Lemma 4.6.2 are exploited for defining a procedure to build the maximal quantifications of CDC single-tile relations over an imprecise reference object with empty yolk, as Theorem 4.6.3 shows. An example is provided in Fig. 4.8 that depicts the CDC quantifications that consider as reference the imprecise object in Fig 3.8(b).

Theorem 4.6.3. *Let O_2^+ be an imprecise object with empty yolk, the maximal quantifications $A_R^+(O_2^+)$, for any $R \in \mathcal{R}_{CDC}^{ST}$ are:*

$$A_B^+(O_2^+) = A_B^+(O_2^\bullet) \quad (4.13) \quad A_{NE}^+(O_2^+) = \bigcup_{s \in BL(O_2^\bullet)} A_{NE}^s(s) \quad (4.18)$$

$$A_N^+(O_2^+) = \bigcup_{s \in BL(O_2^\bullet)} A_N^s(s) \quad (4.14) \quad A_{SE}^+(O_2^+) = \bigcup_{s \in BL(O_2^\bullet)} A_{SE}^s(s) \quad (4.19)$$

$$A_E^+(O_2^+) = \bigcup_{s \in BL(O_2^\bullet)} A_E^s(s) \quad (4.15) \quad A_{SW}^+(O_2^+) = \bigcup_{s \in BL(O_2^\bullet)} A_{SW}^s(s) \quad (4.20)$$

$$A_S^+(O_2^+) = \bigcup_{s \in BL(O_2^\bullet)} A_S^s(s) \quad (4.16) \quad A_{NW}^+(O_2^+) = \bigcup_{s \in BL(O_2^\bullet)} A_{NW}^s(s) \quad (4.21)$$

$$A_W^+(O_2^+) = \bigcup_{s \in BL(O_2^\bullet)} A_W^s(s) \quad (4.17)$$

Proof. (Eq. 4.13) The maximal quantification for the cardinal direction relation B is computed—from Eq. 4.11 and Eq. 4.12—as $A_B^+(O_2^+) = \bigcup_{O_2^\circ \subseteq O_2^+} A_B^+(O_2^\circ) = \bigcup_{O_2^\circ \subseteq O_2^+} MBR(O_2^\circ)$. Furthermore, from Eq. 3.1, $MBR(O_2^\circ) \subseteq MBR(O_2^\bullet)$, and it is equal if $O_2^\circ = O_2^\bullet$. Hence, $MBR(O_2^\bullet) = \bigcup_{O_2^\circ \subseteq O_2^+} MBR(O_2^\circ) \Rightarrow A_B^+(O_2^+) = MBR(O_2^\bullet) = A_B^+(O_2^\bullet)$.

(Eq. 4.14) First, it is proven $A_N^+(O_2^+) = \bigcup_{O_2^\circ \subseteq O_2^+} A_N^+(O_2^\circ) \subseteq \bigcup_{s \in BL(O_2^\bullet)} A_N^s(s)$. This is equivalent to demonstrate that for each $O_2^\circ \subseteq O_2^\bullet$ there exist $s_1, \dots, s_k \in BL(O_2^\bullet)$ such that $A_N^+(O_2^\circ) \subseteq A_N^s(s_1) \cup$

$\dots \cup A_N^s(s_k)$. From Lemma 4.6.2, $O_2^\bullet \subset \bigcup_{s \in BL(O_2^\bullet)} A_N^s(s) \Rightarrow O_2^\circ \subset \bigcup_{s \in BL(O_2^\bullet)} A_N^s(s)$. This means that $\forall (x_o, y_o) \in O_2^\circ, \exists s \in BL(O_2^\bullet) : (x_o, y_o) \in A_N^s(s)$. Furthermore, by definition $A_N^+(O_2^\circ) = \{(x, y) \in \mathbb{R}^2 \mid \exists (x_o, y_o) \in O_2^\circ : x = x_o \wedge y \geq y_o\}$. Thus, $A_N^+(O_2^\circ) \subseteq \bigcup_{s \in BL(O_2^\bullet)} A_N^s(s)$.

In order to prove that the expressions are equal, it is necessary to prove also that $\bigcup_{s \in BL(O_2^\bullet)} A_N^s(s) \subseteq A_N^+(O_2^+) = \bigcup_{O_2^\circ \subseteq O_2^+} A_N^+(O_2^\circ)$. This is trivially done by considering that by definition $\forall s \in BL(O_2^\bullet)$, $A_N^s(s) = \bigcup_{(x,y) \in s} A_N^+(\{(x,y)\}) \subseteq \bigcup_{O_2^\circ \subseteq O_2^+} A_N^+(O_2^\circ)$.

(Eq. 4.15), (Eq. 4.16), (Eq. 4.17), (Eq. 4.18), (Eq. 4.19), (Eq. 4.20), and (Eq. 4.21) can be proven adopting the same approach as (Eq. 4.14). \square

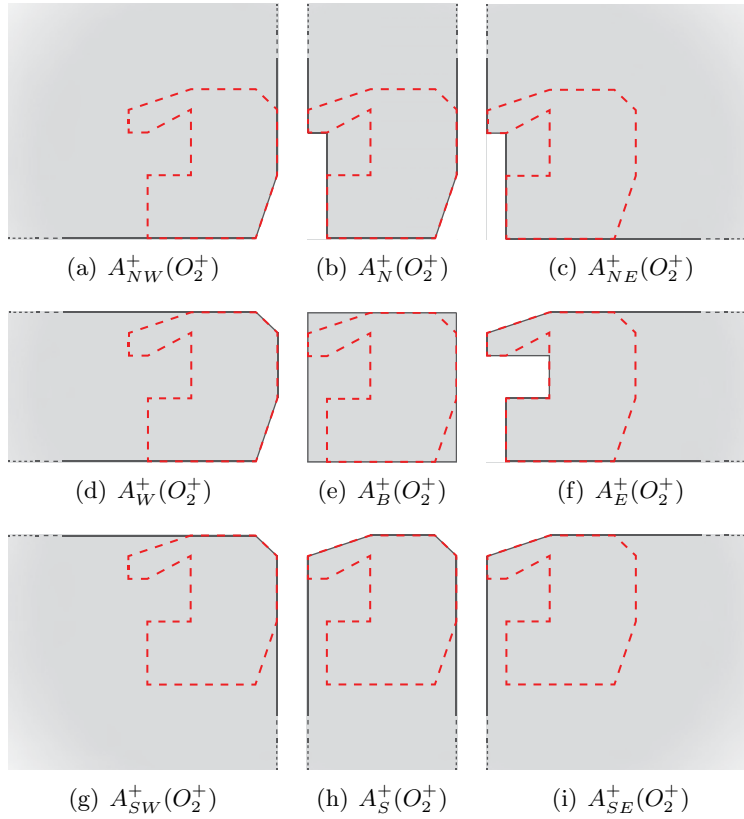


Figure 4.8: Quantification of cardinal direction relations – Case $A_R^+(O_2^+)$.

4.6.1.3 Imprecise Reference Object

Above, the maximal quantifications of CDC relations have been defined by means of infinite-region objects for the cases if the reference object is precise or imprecise with empty yolk. It is now possible to combine the previous outcomes to develop a procedure that computes the maximal quantification of a CDC base single-tile relation if the reference object is imprecise with non-empty yolk. From Equation 3.4 the maximal

quantification for CDC single-tile relations is:

$$A_R^+(O_2^*) = \bigcup_{O_2^- \subseteq O_2^0 \subseteq O_2^+} A_R^+(O_2^0) \quad \text{with } R \in \mathcal{R}_{CDC}^{ST} \quad (4.22)$$

An algorithm for the computation of $A_R^+(O_2^*)$, that is grounded on a combination of the maximal quantifications computed by considering the egg and the yolk objects separately, is shown in Theorem 4.6.4. Furthermore, Fig. 4.9 shows the maximal quantifications for an exemplary imprecise object O_2^* .

Theorem 4.6.4. *Let O_2^* be an imprecise object, the quantifications of the maximal quantifications $A_R^+(O_2^*)$ for $R \in \mathcal{R}_{CDC}^{ST}$ are computed as follows:*

$$A_B^+(O_2^*) = A_B^+(O_2^+) \quad (4.23)$$

$$A_N^+(O_2^*) = A_N^+(O_2^+) \cap (A_N^+(O_2^-) \cup A_{NE}^+(O_2^-) \cup A_{NW}^+(O_2^-)) \quad (4.24)$$

$$A_E^+(O_2^*) = A_E^+(O_2^+) \cap (A_E^+(O_2^-) \cup A_{NE}^+(O_2^-) \cup A_{SE}^+(O_2^-)) \quad (4.25)$$

$$A_S^+(O_2^*) = A_S^+(O_2^+) \cap (A_S^+(O_2^-) \cup A_{SE}^+(O_2^-) \cup A_{SW}^+(O_2^-)) \quad (4.26)$$

$$A_W^+(O_2^*) = A_W^+(O_2^+) \cap (A_W^+(O_2^-) \cup A_{NW}^+(O_2^-) \cup A_{SW}^+(O_2^-)) \quad (4.27)$$

$$A_{NE}^+(O_2^*) = A_{NE}^+(O_2^-) \quad (4.28)$$

$$A_{SE}^+(O_2^*) = A_{SE}^+(O_2^-) \quad (4.29)$$

$$A_{NW}^+(O_2^*) = A_{NW}^+(O_2^-) \quad (4.30)$$

$$A_{SW}^+(O_2^*) = A_{SW}^+(O_2^-) \quad (4.31)$$

Proof. (Eq. 4.23) From Eq. 3.1, for any O_2^0 precise instance of O_2^* , $MBR(O_2^0) \subseteq MBR(O_2^+)$. In particular, $MBR(O_2^0) = MBR(O_2^+)$ if $O_2^0 = O_2^+$. Hence, from Eq. 4.22, $A_B^+(O_2^*) = A_B^+(O_2^+)$.

(Eq. 4.24) Any precise instance O_2^0 of O_2^* has, by definition, to satisfy the following constraints: (1) $O_2^0 \subseteq O_2^+$, and (2) $O_2^- \subseteq O_2^0$. The condition (1) is equivalent to consider O_2^* as an imprecise object with empty yolk, hence Eq. 4.14 holds true and $A_N^+(O_2^0) \subseteq A_N^+(O_2^+)$. The constraint (2), instead, means that $\overline{Y}(O_2^0) \geq \overline{Y}(O_2^-)$ (cf. Eq. 3.1). Hence, from Eq. 4.3, any precise object $o \subseteq \mathbb{R}^2 : N(o, O_2^0) \Rightarrow \overline{Y}(o) \geq \overline{Y}(O_2^0) \geq \overline{Y}(O_2^-)$. In turn, this means that from Eqs. 4.3-4.7-4.10: $o \subseteq (A_N^+(O_2^-) \cup A_{NE}^+(O_2^-) \cup A_{NW}^+(O_2^-))$. Subsequently, $A_N^+(O_2^0)$ is contained in the same union and hence also the union of the quantifications over all the precise instance of O_2^* satisfies the same constraint. Finally, merging (1) and (2), it follows that $A_N^+(O_2^*) = A_N^+(O_2^+) \cap (A_N^+(O_2^-) \cup A_{NE}^+(O_2^-) \cup A_{NW}^+(O_2^-))$.

(Eqs. 4.25-4.26-4.27) The proofs are the same as for Eq. 4.24.

(Eq. 4.28) For any O_2^0 precise instance of O_2^* , the quantification $A_{NE}^+(O_2^0)$ is defined as in Eq. 4.7. That means, $\forall r \subseteq A_{NE}^+(O_2^0) \Rightarrow \underline{X}(r) \geq \overline{X}(O_2^0) \wedge \underline{Y}(r) \geq \overline{Y}(O_2^0)$. From Eq. 3.1, $\overline{X}(O_2^0) \geq \overline{X}(O_2^-) \wedge \overline{Y}(O_2^0) \geq \overline{Y}(O_2^-)$ and hence $\underline{X}(r) \geq \overline{X}(O_2^-) \wedge \underline{Y}(r) \geq \overline{Y}(O_2^-)$. Thus, it follows that $r \subseteq A_{NE}^+(O_2^-) \Rightarrow A_{NE}^+(O_2^0) \subseteq A_{NE}^+(O_2^-)$. In particular, for $O_2^0 = O_2^- \Rightarrow A_{NE}^+(O_2^0) = A_{NE}^+(O_2^-)$. Since $A_{NE}^+(O_2^*)$ is the union of all the quantification computed over all the precise instances of O_2^* (Eq. 4.22), it directly follows $A_{NE}^+(O_2^*) = A_{NE}^+(O_2^-)$.

(Eqs. 4.29-4.30-4.31) The proofs are the same as for Eq. 4.28. \square

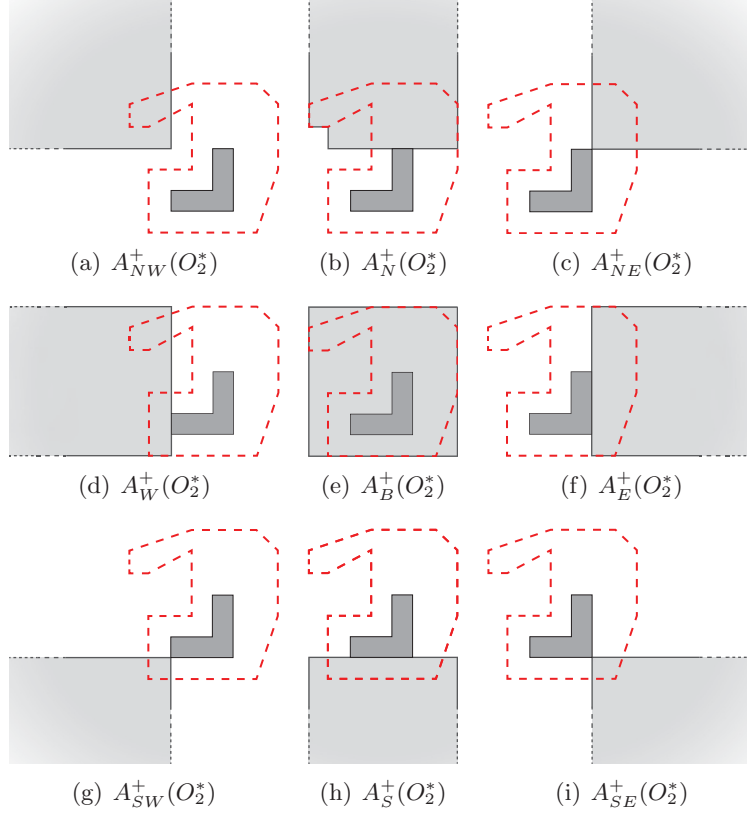


Figure 4.9: Quantification of cardinal direction relations – Case $A_R^+(O_2^*)$.

4.6.1.4 Discussion

The procedure to compute the quantification of a CDC base relation has been described in the above sections. At first, it has been shown that the minimal quantification is always an empty object. In contrast, the procedures for the quantification of CDC single-tile relations have been analyzed for three different cases: precise reference object (Equation 4.11), imprecise reference object with empty yolk (Theorem 4.6.3), and imprecise reference object with non-empty yolk (Theorem 4.6.4). It is not necessary to separately analyze the case in which the reference object's yolk is non-empty while its egg is not defined; indeed, such a case can be modeled as an imprecise object having its egg coincident with \mathbb{R}^2 ; thus, it can be treated as the general case of imprecise reference object with non-empty yolk. An algorithm for the quantification of a CDC base relation (both single and multi-tile) has been shown in Algorithm 4.

The algorithms' outcomes shown above have been proven by considering as reference a spatial-region object that represents a spatial region composed by one component and having finite extent. However, it can be easily shown that the results still hold in the cases if the reference region has infinite extent or if it is composed by more than one component. In the former case the definition of the *MBR* still applies for an infinite-region object (see Appendix A): one or more of the object's maximum and minimum coordinate values are equal to infinite, and the given proofs still hold. The case if the reference object has multiple components is again analyzed case by case: If the reference object is precise, it can not be composed by multiple components. Indeed, in this work it is assumed that the entities in the reality always occupy a single connected region (cf. Section 3.3.3); hence, multi-component precise objects are not admitted and Equation 4.11 already consider all the admitted objects. If a multi-component imprecise object with empty yolk is taken as reference for the relation, from Equation 4.12 it directly follows that the maximal quantification over the object is equivalent to the union of the maximal quantifications build over the object's components separately. Lastly, considering an imprecise object with non-empty yolk as reference, from Axiom 3.3.1 it follows that an imprecise object with non-empty yolk can not have more than one egg component. Furthermore, the case if the yolk is multi-component can be computed considering as yolk the convex-hull of the yolk components.

4.6.2 *Quantify_C* for Visibility Relations

The visibility calculus (Fogliaroni *et al.*, 2009) defines 27 base relations over a domain of convex connected regions in \mathbb{R}^2 , and whose reference objects' convex hulls do not overlap (Section 2.2.3.3). Five relations are single-tile relations while the others are defined as multi-tile ones. As for the CDC, the minimal quantification of every visibility base relation is always an empty object:

$$A_R^-(O_2^*, O_3^*) = \bigcap_{O \subseteq \mathbb{R}^2 \mid R(O, O_2^*, O_3^*)} O = \emptyset \quad \text{with } R \in \mathcal{R}_{Vis}$$

The result is directly proven by considering the objects in Fig. 4.10. Both O'_1 and O''_1 satisfy the relation V with respect to O_2 and O_3 . Furthermore, $O'_1 \cap O''_1 = \emptyset$ from which it directly follows that $A_V^-(O_2, O_3) = \emptyset$. The same result holds for all the visibility relations. Thus, the quantification of a visibility relation always yields a spatial-region object with empty yolk.

Moreover, the maximal quantification of a single-tile relation corresponds to the acceptance area defined in the calculus, while the maximal quantification of a multi-tile relation is given by the union of the quantifications of the different tiles that compose it.

The $Quantify_C$ for the computation of the quantification of a visibility base relation is shown in Algorithm 5. In the remainder of this section, the case if the reference objects are precise will be analyzed first and a definition of the maximal quantifications by means of infinite-region objects is given. Afterwards, the results will be extended taking into account imprecise reference objects.

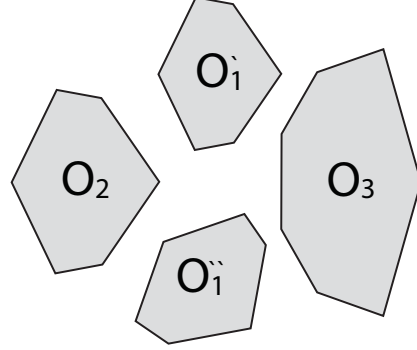


Figure 4.10: $A_V^-(O_2, O_3)$.

Algorithm 5 $Quantify_C(R, O_2^*, O_3^*)$ – Visibility Calculus

```

 $A_R^* \leftarrow (\emptyset, \emptyset)$ 
for  $R^{ST} \in \Gamma(R)$  do
   $A_R^+ \leftarrow A_R^+ \cup A_{R^{ST}}^+(O_2^*, O_3^*)$ 
end for
return  $A_R^*$ 

```

4.6.2.1 Precise Reference Objects

The maximal quantifications of the visibility single-tile relations if the reference objects are precise are equivalent to the relations' acceptance areas. Hence, in order to provide a procedure for quantifying these relations, a representation of the acceptance areas as infinite-region objects is required. In the following, a step-by-step procedure is shown for the definition of the infinite-region object's rays and finite boundaries.

At first, the four common tangents between the reference objects are to be identified; this operation requires beforehand the computation of the two objects' convex hulls. An example of the infinite-region objects construction over two convex objects is shown in Fig. 4.11; for the sake of simplicity, the objects shown in the example are already convex, thus $CH(O_2) = O_2$ and $CH(O_3) = O_3$. The tangents are described by means of the eight tangential points over $CH(O_2)$ and $CH(O_3)$:

$p_{ER}^2 \leftarrow$ point of tangent between ER and $CH(O_2)$ $p_{ER}^3 \leftarrow$ point of tangent between ER and $CH(O_3)$
 $p_{EL}^2 \leftarrow$ point of tangent between EL and $CH(O_2)$ $p_{EL}^3 \leftarrow$ point of tangent between EL and $CH(O_3)$
 $p_{IR}^2 \leftarrow$ point of tangent between IR and $CH(O_2)$ $p_{IR}^3 \leftarrow$ point of tangent between IR and $CH(O_3)$
 $p_{IL}^2 \leftarrow$ point of tangent between IL and $CH(O_2)$ $p_{IL}^3 \leftarrow$ point of tangent between IL and $CH(O_3)$

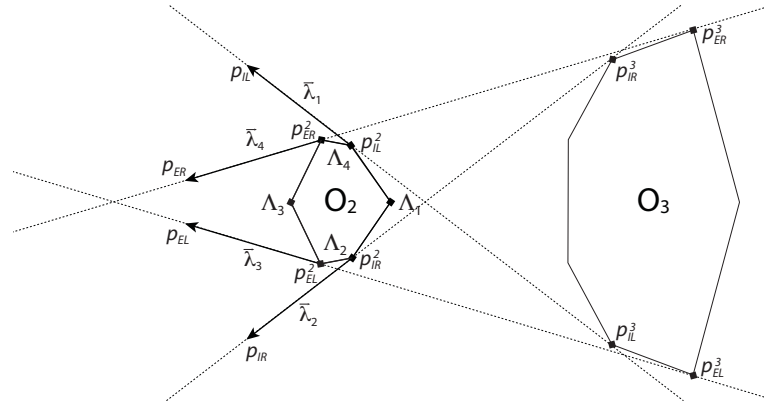


Figure 4.11: Quantification of the visibility relations using infinite-region objects.

The algebraic equations that define the four mutual tangents between $CH(O_2)$ and $CH(O_3)$ —denoted as $f_{ER}(x)$, $f_{EL}(x)$, $f_{IR}(x)$ and $f_{IL}(x)$ —are obtained from the equation of a straight line through two points $p_1 = (x_1, y_1)$ and $p_2 = (x_2, y_2)$, by replacing p_1 and p_2 with the required tangent points:

$$y = f(x) = mx + q \quad \text{with} \quad m = \frac{y_2 - y_1}{x_2 - x_1}, \quad q = y_1 - \frac{y_1 - y_2}{x_1 - x_2}x_2$$

The tangent's equations are used to identify an arbitrary point on every tangent that is on the directed line that connects the point of tangent on $CH(O_3)$ with the point on $CH(O_2)$, and is after the latter point. Those points are denoted respectively with p_{ER} , p_{EL} , p_{IR} and p_{IL} . It is then possible to define the four rays that will be used for building the infinite-region objects:

$$\vec{\lambda}_1 = [p_{IL}^2, p_{IL}] \quad \vec{\lambda}_2 = [p_{IR}^2, p_{IR}] \quad \vec{\lambda}_3 = [p_{EL}^2, p_{EL}] \quad \vec{\lambda}_4 = [p_{ER}^2, p_{ER}]$$

The four tangential points on O_2 always appear in the order $p_{IL}^2, p_{IR}^2, p_{EL}^2$ and p_{ER}^2 into the clockwise ordered list of points that define O_2 . Hence, the polyline

$O_2 = \langle p, \dots, p_{IL}^2, \dots, p_{IR}^2, \dots, p_{EL}^2, \dots, p_{ER}^2, \dots, q \rangle$ can be decomposed into four distinct polylines:

$$\Lambda_1 = \langle p_{IL}^2, \dots, p_{IR}^2 \rangle \quad \Lambda_2 = \langle p_{IR}^2, \dots, p_{EL}^2 \rangle \quad \Lambda_3 = \langle p_{EL}^2, \dots, p_{ER}^2 \rangle \quad \Lambda_4 = \langle p_{ER}^2, \dots, p_{IL}^2 \rangle$$

Finally, the infinite-region objects that define the maximal quantifications of the five visibility single-tile relations are computed as in Equation 4.32. Two auxiliary objects—namely $A_{Le}^+(O_2, O_3)$ and $A_{Ri}^+(O_2, O_3)$ —are used for defining the areas related to the relations PV_L , PV_R , and PV_J . The auxiliary objects have been introduced to build a procedure that computes the different maximal quantifications avoiding to explicitly check whether $A_{PV_J}^+(O_2, O_3)$ exists. Indeed, if the PV_J quantification does not exist, then $A_{PV_L}^+(O_2, O_3) = A_{Le}^+(O_2, O_3)$ and $A_{PV_R}^+(O_2, O_3) = A_{Ri}^+(O_2, O_3)$. Otherwise, the quantification of PV_J corresponds to the intersection of A_{Le}^+ and A_{Ri}^+ , while the quantification of PV_L and PV_R are given by the residual regions.

$$\begin{aligned} A_V^+(O_2, O_3) &= (\Lambda_1, \vec{\lambda}_1, \vec{\lambda}_2) \\ A_{Oc}^+(O_2, O_3) &= (\Lambda_3, \vec{\lambda}_3, \vec{\lambda}_4) & A_{PV_J}^+(O_2, O_3) &= A_{Le}^+(O_2, O_3) \cap A_{Ri}^+(O_2, O_3) \\ A_{Le}^+(O_2, O_3) &= (\Lambda_2, \vec{\lambda}_2, \vec{\lambda}_3) & A_{PV_L}^+(O_2, O_3) &= A_{Le}^+(O_2, O_3) \setminus A_{PV_J}^+(O_2, O_3) \\ A_{Ri}^+(O_2, O_3) &= (\Lambda_4, \vec{\lambda}_4, \vec{\lambda}_1) & A_{PV_R}^+(O_2, O_3) &= A_{Ri}^+(O_2, O_3) \setminus A_{PV_J}^+(O_2, O_3) \end{aligned} \quad (4.32)$$

4.6.2.2 Imprecise Reference Objects

The procedures for the quantification of visibility base relations developed above are exploited in this section for computing the quantification in the case if at least one of the reference objects is imprecise. From Equation 3.4 the maximal quantification of a visibility ternary relation having imprecise reference objects is formally defined as:

$$A_R^+(O_2^*, O_3^*) = \bigcup_{\substack{O_2^- \subseteq O_2^\circ \subseteq O_2^+ \\ O_3^- \subseteq O_3^\circ \subseteq O_3^+}} A_N^+(O_2^\circ, O_3^\circ) \quad \text{with } R \in \mathcal{R}_{Vis}^{ST} \quad (4.33)$$

For any relation, the maximal quantification is computed as a combination of the quantifications over the eggs and the yolks (considered as precise objects) of the reference objects. The combination always yields an object that contains $A_R^+(O_2^*, O_3^*)$.

Any of the reference objects can be either precise, or imprecise with empty yolk, or imprecise with non-empty yolk: eight potential distinct combinations have to be analyzed. The case in which both the objects are precise has been already discussed

above, while the other combinations are summarized in Table 4.1. The equations to compute the maximal quantification of the visibility single-tile relations, and the resulting quantifications for some exemplary reference objects, are shown in Fig. 4.12-4.19¹. The equations can be proven by observing how the tangents vary if different precise instances for the reference objects are considered, and checking which areas build over the eggs and yolks are crossed by the tangents.

Table 4.1: Quantification of Visibility relations with imprecise reference objects.

Obstacle	Viewer	Figure
$O_2^* = (O_2, O_2)$	$O_3^* = (O_3^+, \emptyset)$	Fig. 4.12
$O_2^* = (O_2, O_2)$	$O_3^* = (O_3^+, O_3^-)$	Fig. 4.13
$O_2^* = (O_2^+, \emptyset)$	$O_3^* = (O_3, O_3)$	Fig. 4.14
$O_2^* = (O_2^+, \emptyset)$	$O_3^* = (O_3^+, \emptyset)$	Fig. 4.15
$O_2^* = (O_2^+, \emptyset)$	$O_3^* = (O_3^+, O_3^-)$	Fig. 4.16
$O_2^* = (O_2^+, O_2^-)$	$O_3^* = (O_3, O_3)$	Fig. 4.17
$O_2^* = (O_2^+, O_2^-)$	$O_3^* = (O_3^+, \emptyset)$	Fig. 4.18
$O_2^* = (O_2^+, O_2^-)$	$O_3^* = (O_3^+, O_3^-)$	Fig. 4.19

¹Within the images, the common tangents between O_2^* and a precise object (O_3 or O_3^-) are drawn in black color, while the common tangents between O_2^* and O_3^+ are drawn in red.

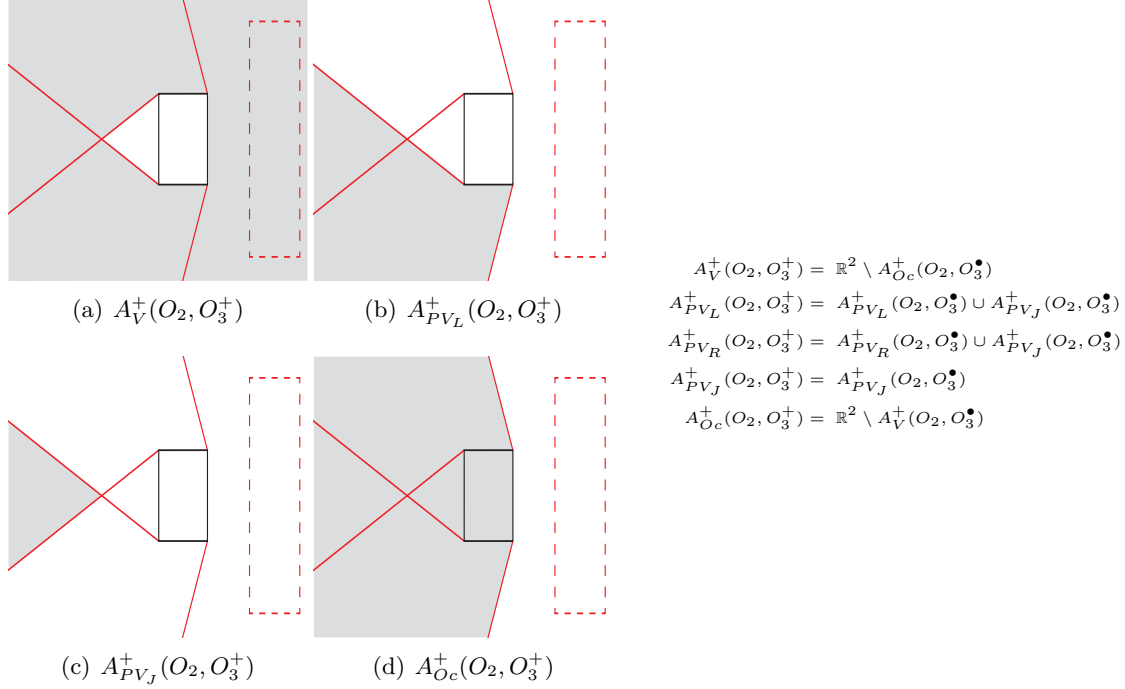


Figure 4.12: $A_{\mathcal{R}_{Vis}^{ST}}^+(O_2, O_3^+)$.

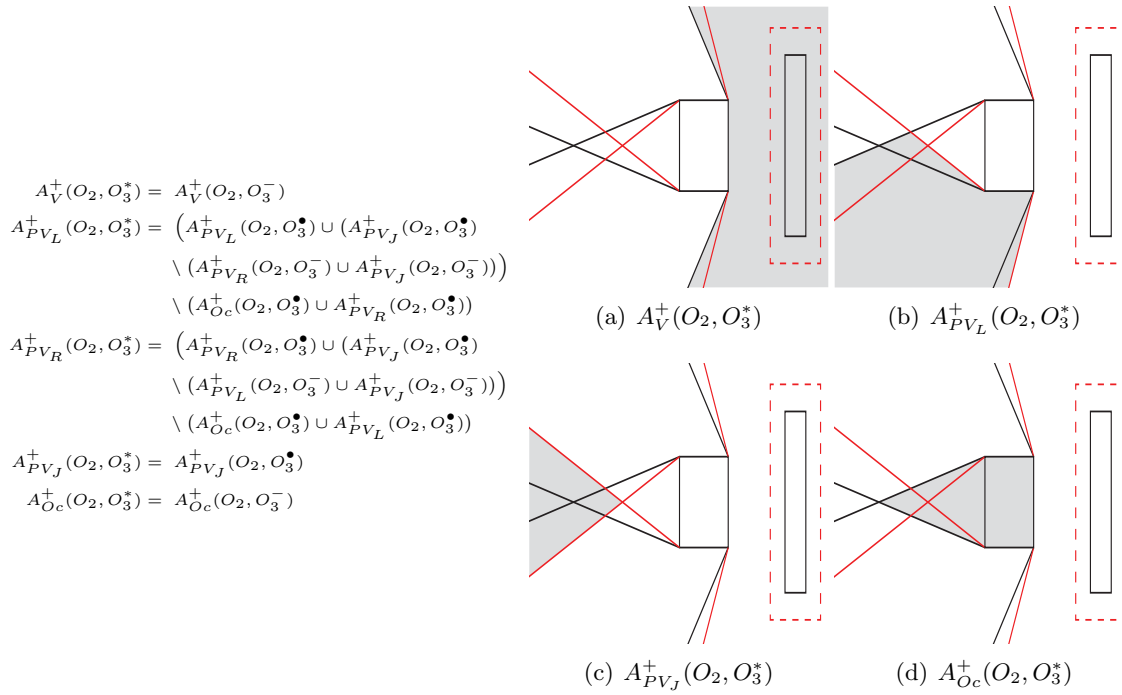


Figure 4.13: $A_{\mathcal{R}_{Vis}^{ST}}^+(O_2, O_3^*)$.

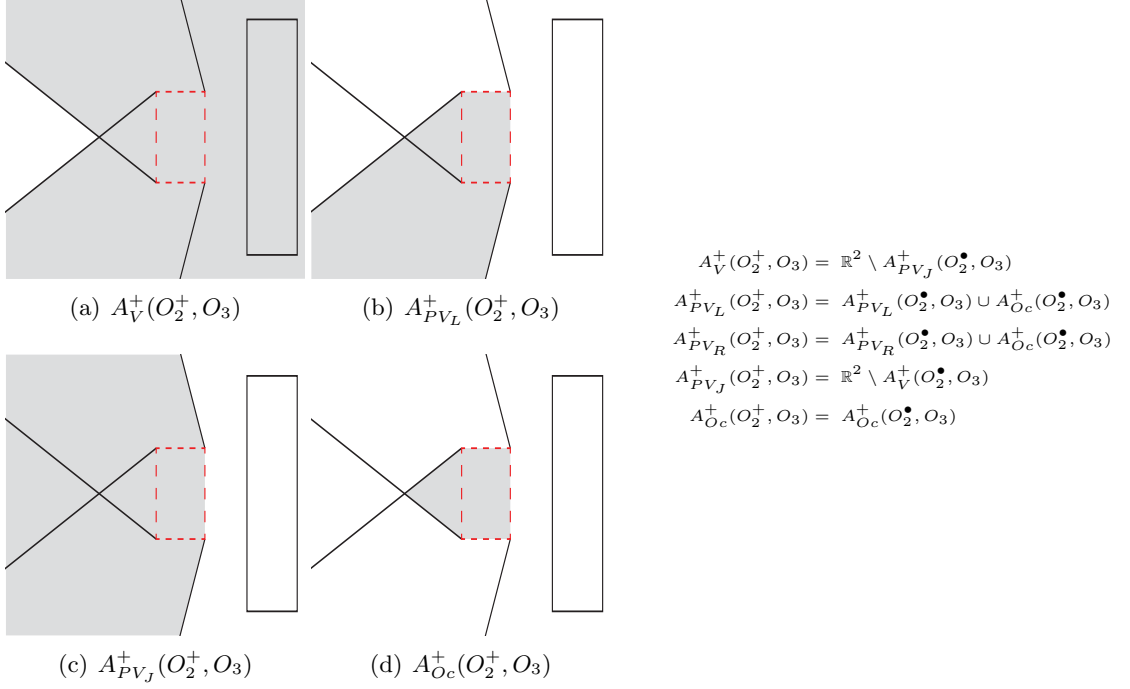


Figure 4.14: $A_{\mathcal{R}_{Vis}^{ST}}^+(O_2^+, O_3)$.

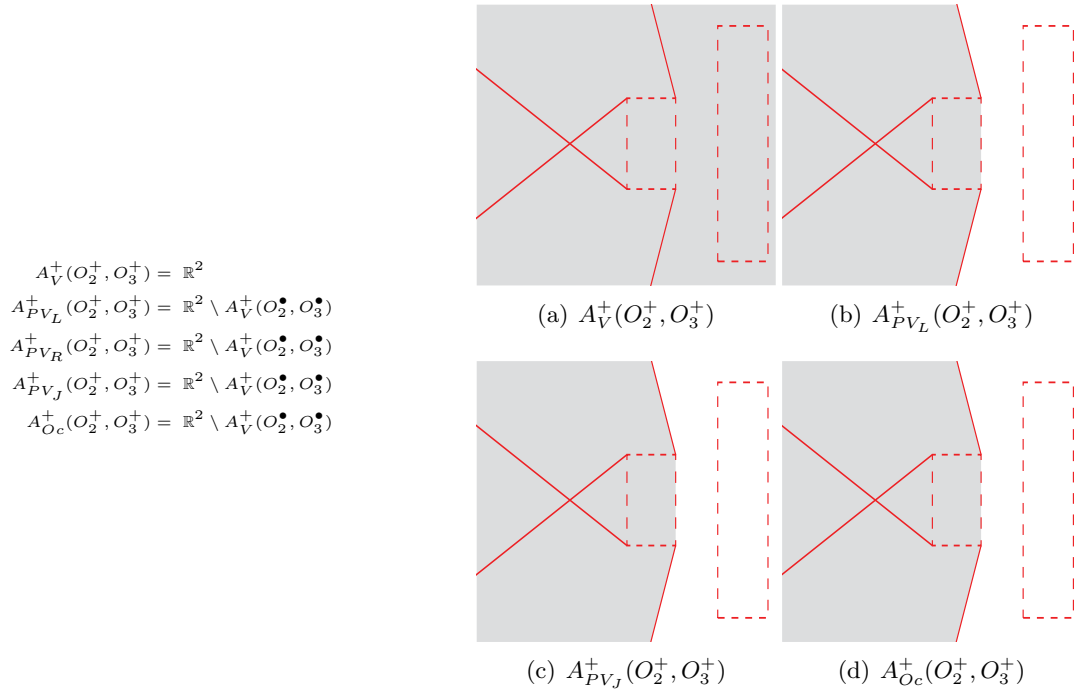


Figure 4.15: $A_{\mathcal{R}_{Vis}^{ST}}^+(O_2^+, O_3^+)$.

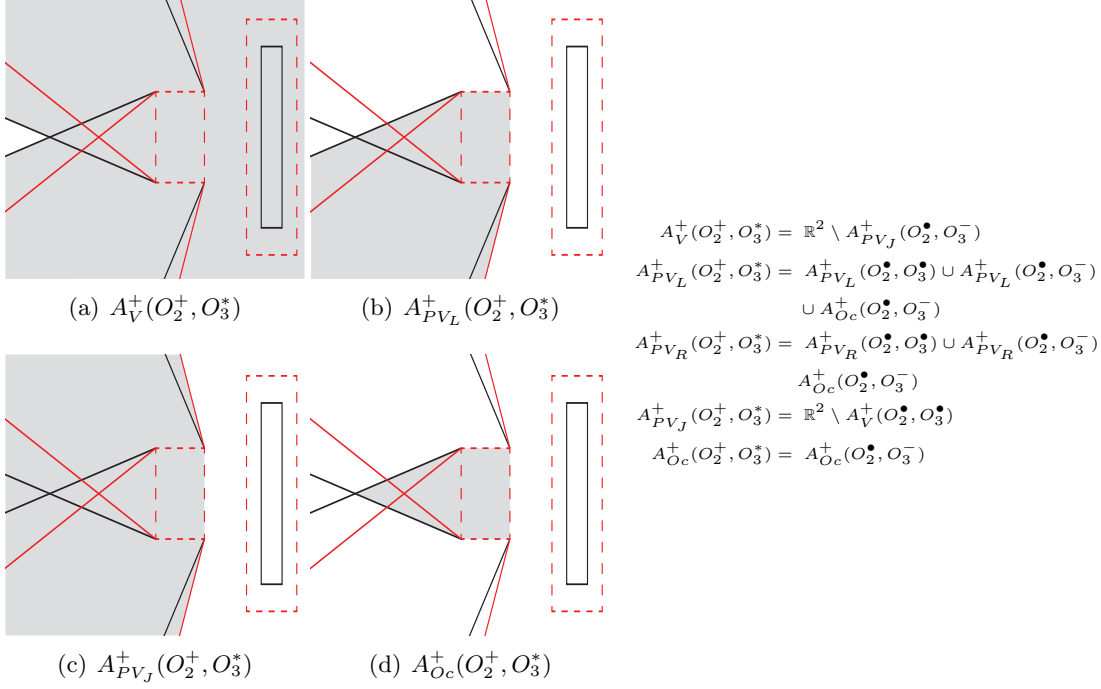


Figure 4.16: $A_{\mathcal{R}_{Vis}^{ST}}^+(O_2^+, O_3^*)$.

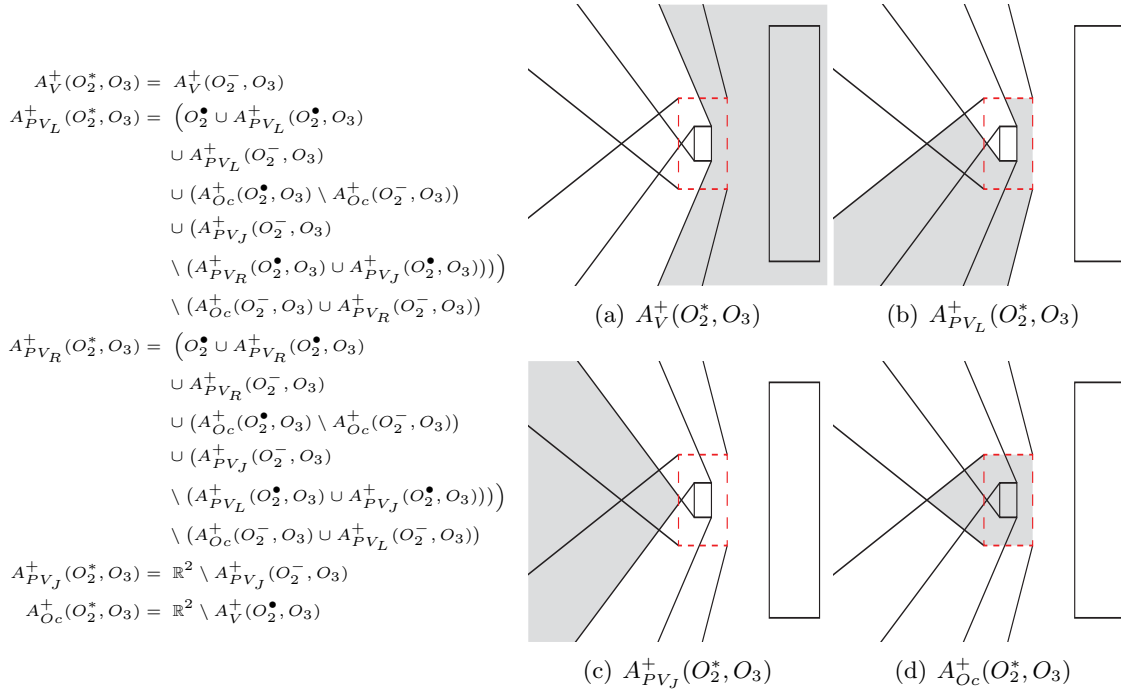
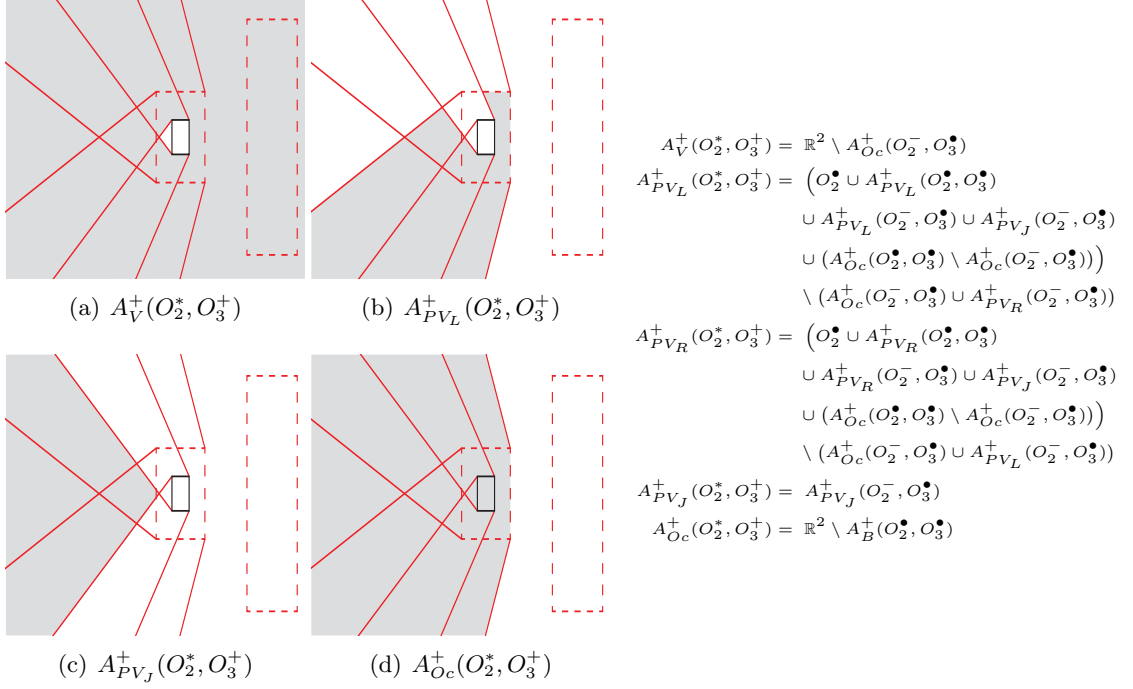
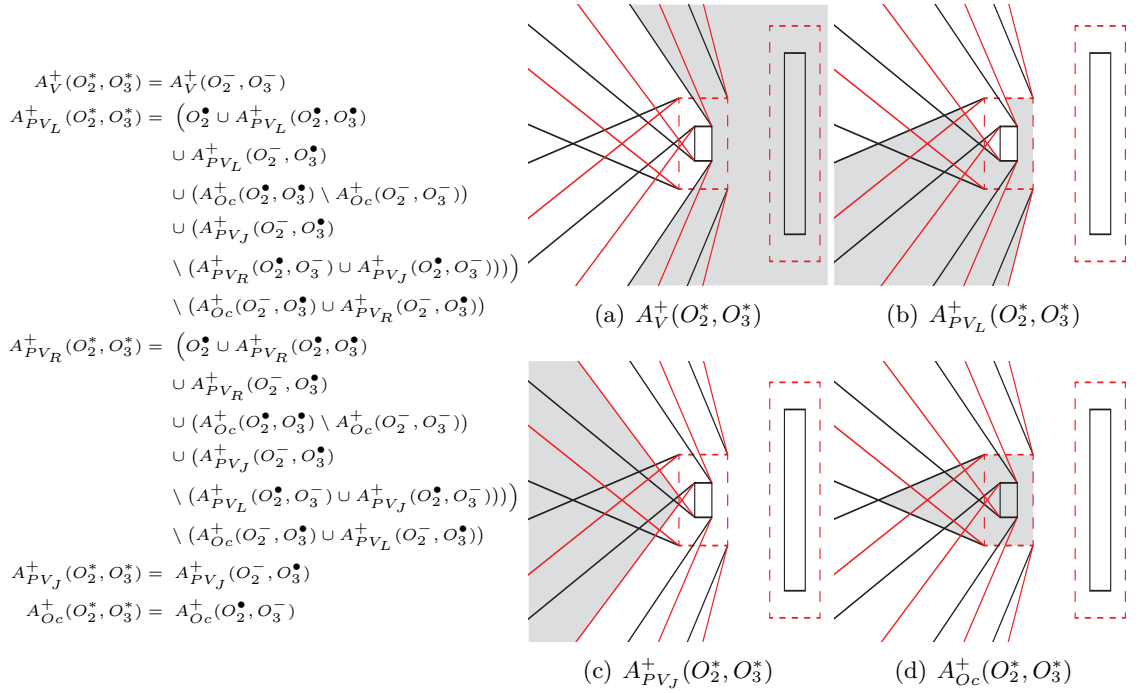


Figure 4.17: $A_{\mathcal{R}_{Vis}^{ST}}^+(O_2^*, O_3)$.

Figure 4.18: $A_{\mathcal{R}_{Vis}^{ST}}^+(O_2^*, O_3^+)$.Figure 4.19: $A_{\mathcal{R}_{Vis}^{ST}}^+(O_2^*, O_3^*)$.

4.6.2.3 Discussion

A quantification procedure for visibility relations has been proposed that is based on the infinite-region representation of the acceptance areas defined in the visibility calculus. In particular, an algorithm to compute the representation in the case of precise reference objects has been given in Eq. 4.32, while Table 4.1 summarizes the different potential combinations of the reference objects. As for the cardinal direction calculus, it is not necessary to analyze the case in which one of the reference objects has non-empty yolk and has not a defined egg: such a case is modeled as an imprecise object having maximal extension coincident with \mathbb{R}^2 .

The quantification of a visibility relation can not be computed for any kind of spatial-region object. Indeed, in the case if at least one of the reference objects represents a region with infinite extent, it is not possible to compute the convex-hull of the object, and also the tangents between the reference objects can not be traced (see Appendix A). Hence, the quantification of a visibility relation having at least one reference object that represents a region with infinite extent is not computable.

4.6.3 *Quantify_C* for Topological Relations

The RCC-8 calculus (Randell *et al.*, 1992) defines eight topological base relations over a domain of connected spatial regions (Section 2.2.3.1). The *Quantify_C* procedure to compute the quantification of a topological relation is shown in Algorithm 6. Differently from cardinal direction and visibility, the minimal quantification of topological relations is not always empty. Moreover, the RCC-8 calculus is not based on the concept of acceptance areas, hence the quantification of topological relation can not be linked to that concept. In the remainder of this section, the quantification procedure if the reference object is precise will be firstly analyzed; afterwards a procedure to compute the quantification in the case of imprecise reference object is given.

Algorithm 6 *Quantify_C*(R, O_2^*) – RCC-8 Calculus

$A_R^+ \leftarrow A_R^+(O_2^*), \quad A_R^- \leftarrow A_R^-(O_2^*)$
return A_R^+, A_R^-

4.6.3.1 Precise Reference Object

The quantification of topological RCC-8 relations having a precise reference object is defined, from Eq. 3.4 and Eq. 3.5, as:

$$A_R^+(O_2) = \bigcup_{O \subseteq \mathbb{R}^2 \mid R(O, O_2)} O \quad \text{with } R \in \mathcal{B}_{RCC} \quad A_R^-(O_2) = \bigcap_{O \subseteq \mathbb{R}^2 \mid R(O, O_2)} O \quad \text{with } R \in \mathcal{B}_{RCC}$$

The properties of the objects that satisfy a RCC-8 relation can be exploited to develop a procedure for the computation of A_R^+ and A_R^- . Considering for instance the relation $TPP(O_1^*, O_2)$ (Fig. 4.20(e)), the primary object O_1^* can lie everywhere inside the reference object, but can not be outside it; hence the maximal quantification of TPP is constrained by the object O_2 . Furthermore, the relation does not give any constraint for the minimal quantification. Conversely, the relation $NTPP_I$ (Fig. 4.20(h)) does not give any constraint for the maximal quantification, but the reference object constraints the minimal quantification since any object that satisfies the relation has to contain O_2 . The quantifications of RCC-8 relations having precise reference object are shown in Fig. 4.20; graphical examples are provided as proof for the equations¹. The results are either infinite-region objects coincident with \mathbb{R}^2 (e.g., $A_{PO}^+(O_2)$, $A_{TPPI}^+(O_2)$, and $A_{NTPP_I}^+(O_2)$), or objects equal to \mathbb{R}^2 having a finite hole (e.g., $A_{DC}^+(O_2)$ and $A_{EC}^+(O_2)$), or spatial-region objects with finite extent.

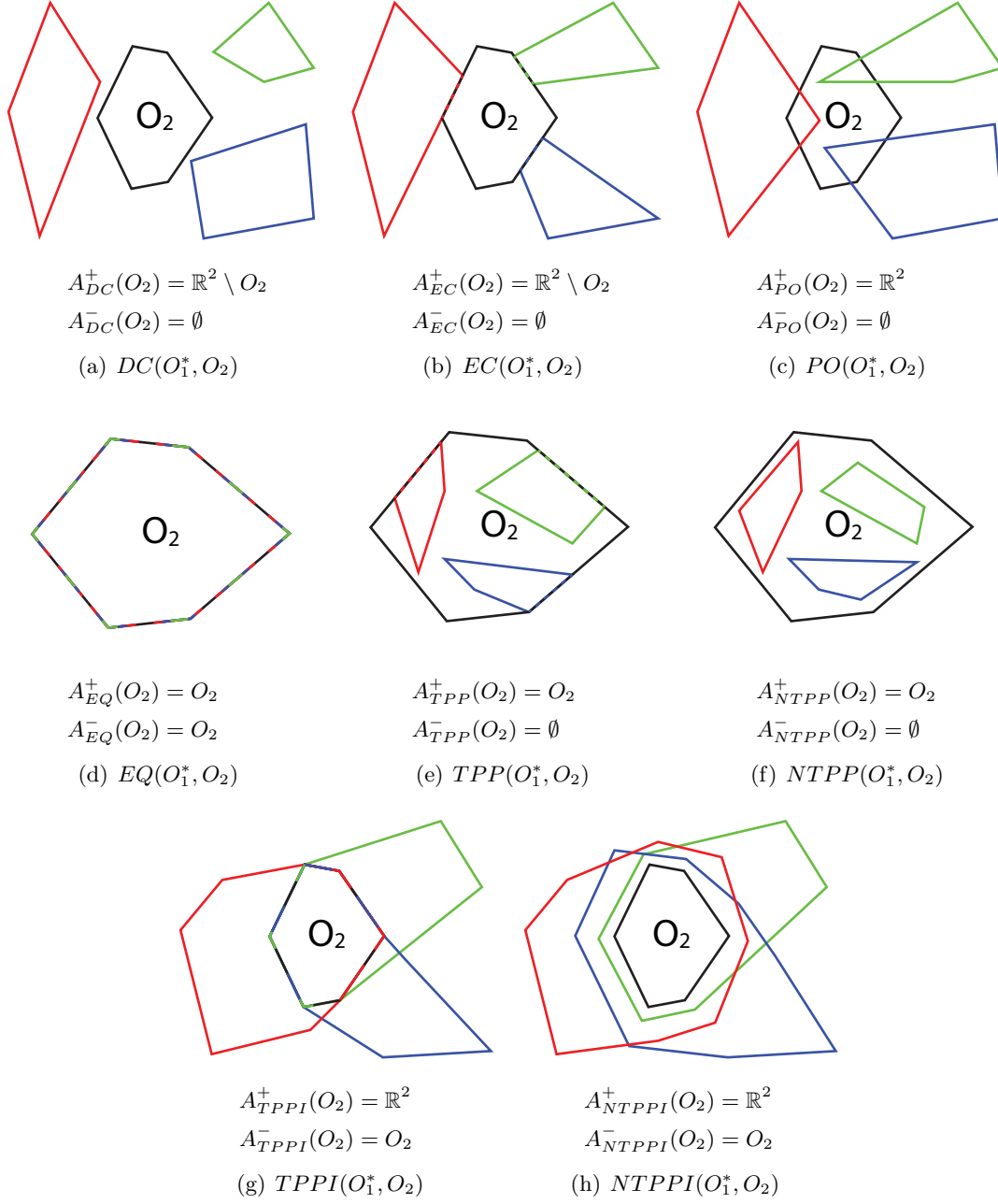
4.6.3.2 Imprecise Reference Object

Given an RCC-8 binary relation having an imprecise reference object, the quantification of the relation—respectively in the cases $O_2^* = (O_2^+, \emptyset)$ and $O_2^* = (O_2^+, O_2^-)$ —is defined, from Eq. 3.4 and Eq. 3.5, as:

$$\begin{aligned}
 A_R^+(O_2^+) &= \bigcup_{O_2^\circ \subseteq O_2^+} A_R^+(O_2^\circ) & \text{with } R \in \mathcal{B}_{RCC} \\
 A_R^-(O_2^+) &= \bigcap_{O_2^\circ \subseteq O_2^+} A_R^-(O_2^\circ) & \text{with } R \in \mathcal{B}_{RCC} \\
 A_R^+(O_2^*) &= \bigcup_{O_2^- \subseteq O_2^\circ \subseteq O_2^+} A_R^+(O_2^\circ) & \text{with } R \in \mathcal{B}_{RCC} \\
 A_R^-(O_2^*) &= \bigcap_{O_2^- \subseteq O_2^\circ \subseteq O_2^+} A_R^-(O_2^\circ) & \text{with } R \in \mathcal{B}_{RCC}
 \end{aligned} \tag{4.34}$$

A procedure to compute the quantification of a topological relation if the reference object is imprecise is given by Theorem 4.6.5. This procedure is valid for any kind of spatial-region object.

¹Within the figures, O_2 is depicted as an object with a black border, while the polygons having different border colors are examples of objects that satisfy the relation with O_2 .

**Figure 4.20:** Quantification of RCC-8 relations.

Theorem 4.6.5. *Given an imprecise object O_2^* , the quantifications $A_R^+(O_2^*)$, $A_R^-(O_2^*)$, $A_R^+(O_2^+)$, and $A_R^-(O_2^+)$ with $R \in \mathcal{B}_{RCC}$ are computed as in Tab. 4.2.*

Table 4.2: Quantification of RCC-8 relations with imprecise reference objects.

R	DC	EC	PO	EQ	TPP	NTPP	TPPI	NTPPI
$A_R^+(O_2^+)$	\mathbb{R}^2	\mathbb{R}^2	\mathbb{R}^2	O_2^\bullet	O_2^\bullet	O_2^\bullet	\mathbb{R}^2	\mathbb{R}^2
$A_R^-(O_2^+)$	\emptyset	\emptyset	\emptyset	\emptyset	\emptyset	\emptyset	\emptyset	\emptyset
$A_R^+(O_2^*)$	$\mathbb{R}^2 \setminus O_2^-$	$\mathbb{R}^2 \setminus O_2^-$	\mathbb{R}^2	O_2^\bullet	O_2^\bullet	O_2^\bullet	\mathbb{R}^2	\mathbb{R}^2
$A_R^-(O_2^*)$	\emptyset	\emptyset	\emptyset	O_2^-	\emptyset	\emptyset	O_2^-	O_2^-

Proof. At first, the proof of the equations $A_R^+(O_2^*)$ is given. Considering the case $R = DC$, from Eq. 4.34, $A_{DC}^+(O_2^*) = \bigcup_{O_2^\circ} A_{DC}^+(O_2^\circ)$ with $O_2^- \subseteq O_2^\circ \subseteq O_2^+$. Furthermore, from the equation in Fig. 4.20(a), $A_{DC}^+(O_2^\circ) = \mathbb{R}^2 \setminus O_2^\circ$. Since $O_2^- \subseteq O_2^\circ$, it follows that $\mathbb{R}^2 \setminus O_2^\circ \supseteq \mathbb{R}^2 \setminus O_2^- \Rightarrow A_{DC}^+(O_2^\circ) \supseteq \mathbb{R}^2 \setminus O_2^-$. Considering $O_2^\circ = O_2^- \Rightarrow A_{DC}^+(O_2^\circ) = \mathbb{R}^2 \setminus O_2^-$ and hence $A_{DC}^+(O_2^*) = \mathbb{R}^2 \setminus O_2^-$. The same proof is valid for $A_{EC}^+(O_2^*)$. For the case $R \in \{PO, TPPI, NTPPI\}$, instead, $A_R^+(O_2^\circ)$ is always equals to \mathbb{R}^2 (Fig. 4.20) and hence the equations in Tab. 4.2 are valid. Finally, considering the case $R = EQ$, from the equation in Fig. 4.20(d), $A_{EQ}^+(O_2^\circ) = O_2^\circ, \forall O_2^- \subseteq O_2^\circ \subseteq O_2^+$. Hence, $A_{EQ}^+(O_2^\circ) \subseteq O_2^\bullet$ and it is equal if $O_2^\circ = O_2^\bullet$. Then, it follows from Eq. 4.34 that $A_{EQ}^+(O_2^*) = O_2^\bullet$. The same proof is valid for $R \in \{TPP, NTPP\}$.

If $R \in \{DC, EC, PO, TPP, NTPP\}$, the equations of the quantification $A_R^-(O_2^*)$ given in Tab. 4.2 are verified since $A_R^-(O_2^\circ) = \emptyset, \forall O_2^- \subseteq O_2^\circ \subseteq O_2^+$ (Fig. 4.20). Conversely, considering the case $R = EQ$, the quantification $A_{EQ}^-(O_2^\circ) = O_2^\circ, \forall O_2^- \subseteq O_2^\circ \subseteq O_2^+$. Hence, $A_{EQ}^-(O_2^\circ) \subseteq O_2^-$ and $A_{EQ}^-(O_2^\circ) = O_2^- \Leftrightarrow O_2^\circ = O_2^-$. Thus, it directly follows from Eq. 4.34 that $A_{EQ}^-(O_2^*) = O_2^-$. The same proof is valid for $R \in \{TPPI, NTPPI\}$.

The equations $A_R^+(O_2^+)$, $A_R^-(O_2^+)$, in the case $O^- = \emptyset$, are obtained from the equations of $A_R^+(O_2^*)$, $A_R^-(O_2^*)$ by imposing an empty yolk object in the results. \square

4.7 Computational Complexity of Quantification

The *Quantify* procedure shown in Algorithm 3 requires the quantification of a variable number of base relations, and the computation of the intersection/union of the results. However, the number of relations to quantify always has an upper bound (8 for RCC-8, 218 for CDC relations, 27 for Visibility). Thus, the computational time complexity of Algorithm 3 is constrained by the computation time required for building the quantifications of direction, visibility, and topological relations in the worst cases. In this section, the complexity of the *Quantify_C* algorithms developed for the distinct qualitative calculi will be analyzed.

4.7.1 Cardinal Direction Quantification

Three different cases have been discussed for the quantification of cardinal direction relations: precise reference object (Equation 4.11), imprecise reference object with empty yolk (Theorem 4.6.3), and imprecise reference object with non-empty yolk (Theorem 4.6.4). The complexity in the three different cases is analyzed separately in order to define the worst case time complexity.

Precise reference object

Let O_2 be a precise object defined by n vertices, the quantification of a CDC single-tile relation only requires the computation of the reference object's MBR (Equation 4.11). All the maximal quantifications are then computed from the points that define the MBR. Hence, the computation of the quantification $A_R^+(O_2)$ requires $\mathcal{O}(n)$ time in the worst case (cf. Section 2.1.2.1).

Imprecise reference object with empty yolk

Let n be the number of vertices that define an imprecise object with empty yolk O_2^+ . If $R = B$, the computational time complexity of $A_B^+(O_2^+)$ (Eq. 4.13) is $\mathcal{O}(n)$ as for the precise reference object case. Conversely, if the relation R is one of the other single-tile relations (Theorem 4.6.3), the computation of $A_R^+(O_2^+)$ requires:

1. The calculation of $n-1$ line segment quantifications $A_R^s(s)$. Any single quantification costs $\mathcal{O}(1)$ time (Lemma 4.6.1). Hence, the overall computational time complexity of this step is $\mathcal{O}(n)$ in the worst case.
2. The union of the $n-1$ quantifications computed in Step 1. In general, the union of two objects defined respectively by m_1 and m_2 vertices yields, in the worst case, an object having $2(m_1 + m_2)$ vertices and the computation costs $\mathcal{O}(m_1 m_2)$ time (Margalit & Knott, 1989). Let m_1, \dots, m_{n-1} be the number of vertices of the $n-1$ objects $A_R^s(s_1), \dots, A_R^s(s_{n-1})$, and $m = \max\{m_1, \dots, m_{n-1}\}$. The i -th union operation yields an object having $\mathcal{O}(2^i m)$ vertices in the worst case. Thus, the $(n-2)$ -th union operation gets as input an object having $\mathcal{O}(2^n m)$ vertices and an object defined by m vertices and runs in $\mathcal{O}(2^n m^2)$ time. The union operation is repeated $n-2$ times, thus the computation of the union of all the quantifications $A_R^s(s)$ requires $\mathcal{O}(n 2^n m^2)$ time in the worst case. However, this result can be simplified considering that $m \leq 4$ (Fig. 4.7(b)). The computational time complexity of this step is hence $\mathcal{O}(n 2^n)$ in the worst case.

Overall, the algorithm proposed for the computation of $A_R^+(O_2^+)$ has a worst case time complexity of $\mathcal{O}(n 2^n)$.

Imprecise reference object

Let n_1 be the number of vertices that define the egg and n_2 be the number of vertices that define the yolk of an imprecise object O_2^* . The quantification of a cardinal direction relation having O_2^* as reference object is computed as in Theorem 4.6.4. The worst case is represented by the quantification of $R \in \{N, E, S, W\}$. In such cases, a quantification over an imprecise object is computed— $\mathcal{O}(n_1 2^{n_1})$ —as well as three quantifications over a precise object— $\mathcal{O}(n_2)$. Since the individual quantification has always a number of vertices less than or equal to 4, the union operation between them requires constant time. Thus, the proposed algorithm for computing the quantification $A_R^+(O_2^*)$ runs in $\mathcal{O}(n_1 2^{n_1} + n_2)$ time in the worst case.

***Quantify_C* for Cardinal Direction Relations**

Let t be the number of tiles that compose the base relation R in Algorithm 4. The quantification of a CDC relation costs $\mathcal{O}(t(n_1 2^{n_1} + n_2))$ time in the worst case. However, t is always less than or equal to 9 and it can be considered as a constant; the computational time complexity in the worst case becomes $\mathcal{O}(n_1 2^{n_1} + n_2)$.

4.7.2 Visibility Quantification

The visibility relations are quantified as in Equation 4.32 if both the reference objects are precise. Differently, if at least one of the reference objects is imprecise, the quantification is performed as summarized in Table 4.1.

Precise reference object

Let n and m be the number of vertices that define respectively the reference objects O_2 and O_3 . The computation of a visibility relation's quantification $A_R^+(O_2, O_3)$ (Equation 4.32) requires the execution of the following steps:

1. Computation of $CH(O_2)$ and $CH(O_3)$. This operation runs in $\mathcal{O}(n + m)$ time (Melkman, 1987).
2. Identification of the mutual tangent points between $CH(O_2)$ and $CH(O_3)$. The computational time complexity of this step is $\mathcal{O}(\log(n + m))$ (Kirkpatrick & Snoeyink, 1995).
3. Definition of the infinite-region object's rays. This step is based on a constant number of algebraic equations and runs in $\mathcal{O}(1)$ time.
4. Definition of the infinite-region object's finite boundary. This step requires an iteration over all the vertices of O_2 and costs $\mathcal{O}(n)$ time.

Hence, the computation of the rays and finite boundaries requires $\mathcal{O}(n + m)$ time in the worst case. The objects $A_V^+(O_2, O_3)$, $A_{O_e}^+(O_2, O_3)$, $A_{L_e}^+(O_2, O_3)$ and $A_{R_i}^+(O_2, O_3)$ are directly computed as in Equation 4.32. Furthermore, any of these objects is defined by a number of vertices that is less or equal to n (cf. Fig. 4.11). Thus, the computation of

$A_{PV_J}^+(O_2, O_3)$, $A_{PV_L}^+(O_2, O_3)$ and $A_{PV_R}^+(O_2, O_3)$ requires $\mathcal{O}(n^2)$ time in the worst case. While $A_{PV_L}^+(O_2, O_3)$ and $A_{PV_R}^+(O_2, O_3)$ are defined by n vertices in the worst case, $A_{PV_J}^+(O_2, O_3)$ is always an object defined by 3 vertices.

Overall, the proposed algorithm for the computation of $A_R^+(O_2, O_3)$ has a worst case time complexity of $\mathcal{O}(n^2 + m)$.

Imprecise reference object

Let n_1, n_2 be the vertices that describe respectively the egg and the yolk of the imprecise object O_2^* , and let m_1, m_2 be the vertices of the egg and the yolk of O_3^* . The quantification of a visibility relation worst case scenario is given by the computation of $A_{PV_L}^+(O_2^*, O_3^*)$ —or equivalently $A_{PV_R}^+(O_2^*, O_3^*)$ (cf. Fig. 4.19). In such a case, nine different visibility quantifications over precise objects have to be computed first. The computational time complexity is $\mathcal{O}(n^2 + m)$, with $n = n_1 + n_2$ and $m = m_1 + m_2$.

Afterwards, a constant number of union and difference operations of objects having either $\mathcal{O}(n_1)$, or $\mathcal{O}(n_2)$ or $\mathcal{O}(n_1 + n_2)$ vertices have to be performed. The union and difference operations require, at all, $\mathcal{O}(n^2)$ time.

Overall, the described algorithm for computing $A_R^+(O_2^*, O_3^*)$ has a worst case time complexity of $\mathcal{O}(n^2 + m)$.

Quantify_C for Visibility Relations

Let t be the number of tiles that compose the base visibility relation R in Algorithm 5. The quantification of a visibility relation costs $\mathcal{O}(t(n^2 + m))$ time in the worst case. However, t is always less than or equal to 5; it can hence be considered as a constant and the worst case computational time complexity is $\mathcal{O}(n^2 + m)$.

4.7.3 Topology Quantification

The quantification of topological relations are performed by the equations described in Fig. 4.20 if the reference object is precise, otherwise they are performed by the equations listed in Table 4.2. The quantifications are always either equal to \mathbb{R}^2 , or empty, or equal to the egg or the yolk of the reference object, or equivalent to \mathbb{R}^2 with a hole corresponding to the yolk of the reference object. Hence, the computation time complexity is always $\mathcal{O}(1)$ in the topological case.

4.8 Summary

In this chapter, a system for qualitative and quantitative spatial information translation has been described that performs quantification operations of qualitative spatial relations, and qualification operations from the geometries that describe the spatial entities. The focus has then been moved to the development of the quantification component.

As shown by Wolter & Wallgrün (2012), quantification still represents a challenge in the QSR field. Here, a general algorithm for the computation of the quantification has been described that is valid for any qualitative spatial calculus. Specialized quantification algorithms have been described for topological, cardinal direction and visibility relations. The algorithms' outcomes are compatible with the acceptance areas concept used by the visibility and CDC calculi in the case if the reference objects are precise. The outcomes have been extended to regions imprecisely described. Furthermore, the proposed approach works also for calculi that do not define acceptance areas, as the topological calculus.

Chapter 5

Qualification of Quantitative Spatial Information

In the previous chapter a system for the translation of qualitative information to quantitative and vice versa has been described. The system is grounded on two main components: the quantification component and the qualification component. The former has been analyzed in the previous chapter, while the latter will be discussed in the present chapter. A description of the qualification problem will be given in Section 5.1. A procedure for the qualification of topological relations is described in Section 5.2. Section 5.3 analyzes a straightforward qualification approach for visibility relations. Section 5.4 shows how the approach developed for visibility relations can be directly applied to cardinal direction relations and, in general, to other projective calculi. Subsequently, a strategy that exploits the structure of cardinal direction relations is shown to reduce the complexity of the qualification process. Finally, the analysis of the computational complexity of the described algorithms is discussed in Section 5.5.

5.1 The Qualification Algorithm

As has been shown in Section 4.1, the *qualification* operation (i.e., the translation from quantitative to qualitative spatial information) gets as input an n -ary calculus C defined over a domain of connected regions in \mathbb{R}^2 and a set of n spatial-region objects. It yields the qualitative relation, defined in the calculus, that holds between the objects. Thus, a calculus-dependent function $Qualify_C(O_1^*, \dots, O_n^*)$ is defined that qualifies the relation in the calculus C between the spatial-region objects O_1^*, \dots, O_n^* .

If the input objects are precise, the qualification operation basically only requires to check the constraints that a certain calculus defines for the objects in a specific relation. For instance, to qualify a topological relation between two precise objects it

is necessary to check whether the objects are connected, while for cardinal direction single-tile relations it is sufficient to apply the constraints described by Skiadopoulos & Koubarakis (2004). However, if the objects are imprecise, the definitions of the relations are not sufficient anymore. The existing research in this direction focused on the definition of *fuzzy relations* between imprecise objects. Namely, a new set of relations is defined that takes into account the uncertainty of the given objects. For that reason, the set of new relations is not the same as the set of relations defined to deal with precise objects. For instance, Cohn & Gotts (1996) defines a set of topological fuzzy relations based on the RCC calculus, while Cicerone & Di Felice (2000) extend the direction relation matrix by considering objects with a broad boundary.

However, the entities considered in this work have a well-defined boundary in the reality, but their description in the system can result to be imprecise. Hence, the focus is not on qualifying the fuzzy relation, but rather to identify which relations are admitted for a set of entities that are imprecisely described. In other words, the qualification operation has to yield the set of relations that are satisfied by at least one precise instance of any input object. An example has been provided in Fig. 3.11 where the precise instances of two imprecise objects can satisfy the topological relations *DC*, *EC* and *PO*.

Let C be an n -ary qualitative spatial calculus defined over a domain of connected regions in \mathbb{R}^2 , and let O_1^*, \dots, O_n^* be spatial-region objects, the disjunctive relation composed by all base relations in C that hold between precise instances of O_1^*, \dots, O_n^* is called *crisp relation between imprecise objects* and is denoted as $R_C(O_1^*, \dots, O_n^*)$. For the example described above, $R_{RCC}(O_1^*, O_2^*) = \{DC, EC, PO\}$. The qualification problem has been discussed by Wolter & Wallgrün (2012) that identify two main challenges: consistently mapping floating point values to qualitative labels, and devise efficient algorithms that enumerate all possible base relations that can hold between objects imprecisely described. This work aims at tackling the latter challenge, hence a set of functions $Qualify_C$ are developed for the efficient computation of $R_C(O_1^*, \dots, O_n^*)$.

In the remainder of this chapter, the qualification process of the crisp relation between imprecise objects for the topological, visibility, and cardinal direction calculi defined in Chapter 2 will be analyzed.

5.2 Qualification of Topological Relations

Given two imprecise spatial-region objects, the qualification of the topological crisp disjunctive relation between them is defined, from Definition 12, as:

Definition 15 (Topological crisp relation between imprecise objects).

Let O_1^* and O_2^* be two imprecise spatial-region objects, the disjunctive topological relation

that describes the possible relations holding between any precise instance of O_1^* and any precise instance of O_2^* , called topological crisp relation between imprecise objects, is defined as:

$$R_{RCC}(O_1^*, O_2^*) \triangleq \left\{ R \in \mathcal{B}_{RCC} \mid \exists O_1^\circ, \exists O_2^\circ \text{ s.t. } R(O_1^\circ, O_2^\circ) \right\}$$

However, as has been shown in Section 3.3.5, this definition is based on an infinite number of precise objects and it does not provide a constructive procedure to identify the topological crisp relation holding between two imprecise objects. Therefore, a procedure to retrieve the disjunctive relation from the topological relations holding between the components of the primary object—namely the egg interpreted as a precise object and the yolk—and the components of the reference object is required.

The set of topological relations that can hold between the precise instances of two imprecise objects has been adopted in Cohn & Gotts (1996) as a parameter to cluster the fuzzy relations between imprecise objects. However, Cohn & Gotts (1996) only consider the relations defined in the RCC-5 calculus and hence the full set of relations defined by the RCC-8 calculus is not taken into account. Furthermore, they assume that an imprecise object always has well-defined egg and yolk components. This is in contrast with the assumptions of this work in which even imprecise objects with empty yolks are admitted.

In order to develop a procedure to compute the topological crisp relation between two spatial-region objects, the topological properties of the egg-yolk approach can be exploited. For instance, given a precise object O_1 and an imprecise object with empty yolk O_2^+ , any precise instance of O_2^+ has to satisfy the condition $O_2^\circ \subseteq O_2^+ \Leftrightarrow O_2^\circ \subseteq O_2^\bullet$, that in topological terms can be expressed as $\{EQ, TPP, NTPP\} (O_2^\circ, O_2^\bullet)$. To check whether a certain topological relation between O_1 and a precise instance of O_2^+ is allowed, a constraint network that describes the constraints between O_2° and O_2^\bullet and the relation to be checked between O_1 and O_2° can be tested against consistency: if the network is consistent, the relation is admissible, otherwise it is not. As an example, Fig. 5.1 shows the constraint networks for two possible relations between O_1 and O_2° in the case $DC(O_1, O_2^\bullet)$. The black edges represent the relations that have been set to the network, while the red edges are the relations inferred by the propagation of the constraints using the composition table defined in Section 2.2.3.1. In Fig. 5.1(a), the relation $DC(O_1, O_2^\circ)$ is checked; it results that $R(O_1, O_2^\circ) = R(O_1, O_2^\bullet) \diamond R(O_2^\bullet, O_2^\circ) = DC \diamond \{EQ, TPPI, NTPPI\} = DC$ and the network is consistent. Hence, a precise instance of O_2^+ such that $DC(O_1, O_2^\circ)$ exists and the relation DC is contained in $R_{RCC}(O_1, O_2^+)$. In contrast, in Fig. 5.1(b) the relation $EC(O_1, O_2^\circ)$ is checked; in this case the composition operation results in $R(O_1, O_2^\bullet) = EC \diamond \{EQ, TPPI, NTPPI\} =$

DC that creates an inconsistency in the network. Hence, the relation EC is not an element of the topological crisp relation among imprecise objects $R_{RCC}(O_1, O_2^+)$.

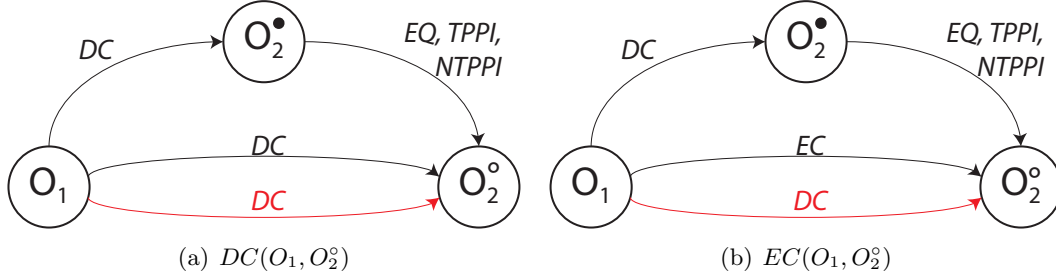


Figure 5.1: Consistency checking of $R(O_1, O_2^o)$.

Therefore, the set of possible relations that can hold between any precise instance of two imprecise objects O_1^* and O_2^* can be computed by defining a constraint network that describes: (1) the relation between the egg (respectively the yolk) of the imprecise objects and any precise instance of the object itself, (2) the relations between the eggs (interpreted as precise) and the yolks of both objects, and (3) the relation that has to be checked. For an imprecise object O^* , the constraints to impose are $\{EQ, TPPI, NTPPI\}(O^o, O^-)$ and $\{EQ, TPP, NTPP\}(O^o, O^*)$.

All the relations have been checked performing the algebraic closure algorithm (Mackworth, 1977) implemented in the qualitative spatial reasoning toolbox SparQ (Wallgrün *et al.*, 2007). Nine different cases can occur, based on whether the objects are precise, imprecise with empty yolk or imprecise with non-empty yolk. The case in which both the objects are precise is the one described by the RCC-8 calculus. Furthermore, some of the cases can be computed as inverse of other relations: $R_{RCC}(O_1^+, O_2) = R_{RCC}(O_2, O_1^+)$, $R_{RCC}(O_1^+, O_2^+) = R_{RCC}(O_2^+, O_1^+)$, and $R_{RCC}(O_1^*, O_2^+) = R_{RCC}(O_2^+, O_1^*)$. Hence only the cases $R_{RCC}(O_1, O_2^+)$, $R_{RCC}(O_1, O_2^*)$, $R_{RCC}(O_1^+, O_2^+)$, $R_{RCC}(O_1^+, O_2^*)$, and $R_{RCC}(O_1^*, O_2^*)$ have to be computed.

The results for the case if both objects are imprecise and have non-empty yolk are shown in Tab. 5.1, while the other cases are shown in Tab. 5.2.

Table 5.1: Qualification of RCC-8 relations – $R_{RCC}(O_1^*, O_2^*)$.

$R(O_1^*, O_2^*)$	$R(O_1^-, O_2^-)$	$R(O_1^*, O_2^-)$	$R(O_1^-, O_2^*)$	$R(O_1^*, O_2^*)$
DC	DC	DC	DC	DC
EC	DC	DC	DC	DC, EC
EC	DC	DC	EC	DC, EC
EC	DC	EC	DC	DC, EC
EC	DC	EC	EC	DC, EC
EC	EC	EC	EC	EC
PO	DC	DC	DC	DC, EC, PO
PO	DC	DC	EC	DC, EC, PO
PO	DC	DC	PO	DC, EC, PO
PO	DC	DC	$TPPI$	DC, EC, PO, TPP
PO	DC	DC	$NTPPI$	$DC, EC, NTPP, PO, TPP$

$R(O_1^*, O_2^*)$	$R(O_1^-, O_2^-)$	$R(O_1^*, O_2^-)$	$R(O_1^-, O_2^*)$	$R(O_1^*, O_2^*)$
PO	DC	EC	DC	DC, EC, PO
PO	DC	EC	EC	DC, EC, PO
PO	DC	EC	PO	DC, EC, PO
PO	DC	EC	TPPI	DC, EC, PO, TPP
PO	DC	EC	NTPPI	DC, EC, NTPP, PO, TPP
PO	DC	PO	DC	DC, EC, PO
PO	DC	PO	EC	DC, EC, PO
PO	DC	PO	PO	DC, EC, PO
PO	DC	PO	TPPI	DC, EC, PO, TPP
PO	DC	PO	NTPPI	DC, EC, NTPP, PO, TPP
PO	DC	TPPI	DC	DC, EC, PO, TPPI
PO	DC	TPPI	EC	DC, EC, PO, TPPI
PO	DC	TPPI	PO	DC, EC, PO, TPPI
PO	DC	TPPI	TPPI	DC, EC, EQ, PO, TPP, TPPI
PO	DC	TPPI	NTPPI	$\mathcal{B}_{RCC} \setminus \{NTPPI\}$
PO	DC	NTPPI	DC	DC, EC, NTPPI, PO, TPPI
PO	DC	NTPPI	EC	DC, EC, NTPPI, PO, TPPI
PO	DC	NTPPI	PO	DC, EC, NTPPI, PO, TPPI
PO	DC	NTPPI	TPPI	$\mathcal{B}_{RCC} \setminus \{NTPP\}$
PO	DC	NTPPI	NTPPI	\mathcal{B}_{RCC}
PO	EC	EC	EC	EC, PO
PO	EC	EC	PO	EC, PO
PO	EC	EC	TPPI	EC, PO, TPP
PO	EC	EC	NTPPI	EC, NTPP, PO, TPP
PO	EC	PO	EC	EC, PO
PO	EC	PO	PO	EC, PO
PO	EC	PO	TPPI	EC, PO, TPP
PO	EC	PO	NTPPI	EC, NTPP, PO, TPP
PO	EC	TPPI	EC	EC, PO, TPPI
PO	EC	TPPI	PO	EC, PO, TPPI
PO	EC	TPPI	TPPI	EC, EQ, PO, TPP, TPPI
PO	EC	TPPI	NTPPI	EC, EQ, NTPP, PO, TPP, TPPI
PO	EC	NTPPI	EC	EC, NTPPI, PO, TPPI
PO	EC	NTPPI	PO	EC, NTPPI, PO, TPPI
PO	EC	NTPPI	TPPI	EC, EQ, NTPPI, PO, TPP, TPPI
PO	EC	NTPPI	NTPPI	$\mathcal{B}_{RCC} \setminus \{DC\}$
PO	PO	PO	PO	PO
PO	PO	PO	TPPI	PO, TPP
PO	PO	PO	NTPPI	NTPP, PO, TPP
PO	PO	TPPI	PO	PO, TPPI
PO	PO	TPPI	TPPI	EQ, PO, TPP, TPPI
PO	PO	TPPI	NTPPI	EQ, NTPP, PO, TPP, TPPI
PO	PO	NTPPI	PO	NTPPI, PO, TPPI
PO	PO	NTPPI	TPPI	EQ, NTPPI, PO, TPP, TPPI
PO	PO	NTPPI	NTPPI	$\mathcal{B}_{RCC} \setminus \{DC\}$
PO	EQ	TPPI	TPPI	EQ, PO, TPP, TPPI
PO	EQ	TPPI	NTPPI	EQ, NTPP, PO, TPP, TPPI
PO	EQ	NTPPI	TPPI	EQ, NTPPI, PO, TPP, TPPI
PO	EQ	NTPPI	NTPPI	$\mathcal{B}_{RCC} \setminus \{DC\}$
PO	TPP	PO	TPPI	PO, TPP
PO	TPP	PO	NTPPI	NTPP, PO, TPP
PO	TPP	TPPI	TPPI	EQ, PO, TPP, TPPI
PO	TPP	TPPI	NTPPI	EQ, NTPP, PO, TPP, TPPI
PO	TPP	NTPPI	TPPI	EQ, NTPPI, PO, TPP, TPPI
PO	TPP	NTPPI	NTPPI	$\mathcal{B}_{RCC} \setminus \{DC\}$
PO	NTPP	PO	NTPPI	EC, NTPP, PO, TPP
PO	NTPP	TPPI	NTPPI	EC, EQ, NTPP, PO, TPP, TPPI
PO	NTPP	NTPPI	NTPPI	$\mathcal{B}_{RCC} \setminus \{DC\}$
PO	TPPI	TPPI	PO	PO, TPPI
PO	TPPI	TPPI	TPPI	EQ, PO, TPP, TPPI
PO	TPPI	TPPI	NTPPI	EQ, NTPP, PO, TPP, TPPI
PO	TPPI	NTPPI	PO	NTPPI, PO, TPPI
PO	TPPI	NTPPI	TPPI	EQ, NTPPI, PO, TPP, TPPI
PO	TPPI	NTPPI	NTPPI	$\mathcal{B}_{RCC} \setminus \{DC\}$
PO	NTPPI	NTPPI	PO	EC, NTPPI, PO, TPPI
PO	NTPPI	NTPPI	TPPI	EC, EQ, NTPPI, PO, TPP, TPPI
PO	NTPPI	NTPPI	NTPPI	$\mathcal{B}_{RCC} \setminus \{DC\}$
EQ	DC	TPPI	TPPI	DC, EC, EQ, PO, TPP, TPPI
EQ	DC	TPPI	NTPPI	$\mathcal{B}_{RCC} \setminus \{NTPPI\}$

$R(O_1^\bullet, O_2^\bullet)$	$R(O_1^-, O_2^-)$	$R(O_1^\bullet, O_2^-)$	$R(O_1^-, O_2^\bullet)$	$R(O_1^\bullet, O_2^\bullet)$
EQ	DC	NTPPI	TPPI	$\mathcal{B}_{RCC} \setminus \{NTPP\}$
EQ	DC	NTPPI	NTPPI	\mathcal{B}_{RCC}
EQ	EC	TPPI	TPPI	EC, EQ, PO, TPP, TPPI
EQ	EC	TPPI	NTPPI	EC, EQ, NTPP, PO, TPP, TPPI
EQ	EC	NTPPI	TPPI	EC, EQ, NTPPI, PO, TPP, TPPI
EQ	EC	NTPPI	NTPPI	$\mathcal{B}_{RCC} \setminus \{DC\}$
EQ	PO	TPPI	TPPI	EQ, PO, TPP, TPPI
EQ	PO	TPPI	NTPPI	EQ, NTPP, PO, TPP, TPPI
EQ	PO	NTPPI	TPPI	EQ, NTPPI, PO, TPP, TPPI
EQ	PO	NTPPI	NTPPI	$\mathcal{B}_{RCC} \setminus \{DC\}$
EQ	EQ	EQ	EQ	EQ
EQ	EQ	TPPI	TPPI	EQ, PO, TPP, TPPI
EQ	EQ	NTPPI	NTPPI	$\mathcal{B}_{RCC} \setminus \{DC\}$
EQ	TPP	EQ	TPPI	EQ, TPP
EQ	TPP	TPPI	TPPI	EQ, PO, TPP, TPPI
EQ	TPP	TPPI	NTPPI	EQ, NTPP, PO, TPP, TPPI
EQ	TPP	NTPPI	NTPPI	$\mathcal{B}_{RCC} \setminus \{DC\}$
EQ	NTPP	EQ	NTPPI	EQ, NTPP, TPP
EQ	NTPP	TPPI	NTPPI	EC, EQ, NTPP, PO, TPP, TPPI
EQ	NTPP	NTPPI	NTPPI	$\mathcal{B}_{RCC} \setminus \{DC\}$
EQ	TPPI	TPPI	EQ	EQ, TPPI
EQ	TPPI	TPPI	TPPI	EQ, PO, TPP, TPPI
EQ	TPPI	NTPPI	TPPI	EQ, NTPPI, PO, TPP, TPPI
EQ	TPPI	NTPPI	NTPPI	$\mathcal{B}_{RCC} \setminus \{DC\}$
EQ	NTPPI	NTPPI	EQ	EQ, NTPPI, TPPI
EQ	NTPPI	NTPPI	TPPI	EC, EQ, NTPPI, PO, TPP, TPPI
EQ	NTPPI	NTPPI	NTPPI	$\mathcal{B}_{RCC} \setminus \{DC\}$
TPP	DC	DC	TPPI	DC, EC, PO, TPP
TPP	DC	DC	NTPPI	DC, EC, NTPP, PO, TPP
TPP	DC	EC	TPPI	DC, EC, PO, TPP
TPP	DC	EC	NTPPI	DC, EC, NTPP, PO, TPP
TPP	DC	PO	TPPI	DC, EC, PO, TPP
TPP	DC	PO	NTPPI	DC, EC, NTPP, PO, TPP
TPP	DC	TPPI	TPPI	DC, EC, EQ, PO, TPP, TPPI
TPP	DC	TPPI	NTPPI	$\mathcal{B}_{RCC} \setminus \{NTPPI\}$
TPP	DC	NTPPI	TPPI	$\mathcal{B}_{RCC} \setminus \{NTPP\}$
TPP	DC	NTPPI	NTPPI	\mathcal{B}_{RCC}
TPP	EC	EC	TPPI	EC, PO, TPP
TPP	EC	EC	NTPPI	EC, NTPP, PO, TPP
TPP	EC	PO	TPPI	EC, PO, TPP
TPP	EC	PO	NTPPI	EC, NTPP, PO, TPP
TPP	EC	TPPI	TPPI	EC, EQ, PO, TPP, TPPI
TPP	EC	TPPI	NTPPI	EC, EQ, NTPP, PO, TPP, TPPI
TPP	EC	NTPPI	TPPI	EC, EQ, NTPPI, PO, TPP, TPPI
TPP	EC	NTPPI	NTPPI	$\mathcal{B}_{RCC} \setminus \{DC\}$
TPP	PO	PO	TPPI	PO, TPP
TPP	PO	PO	NTPPI	NTPP, PO, TPP
TPP	PO	TPPI	TPPI	EQ, PO, TPP, TPPI
TPP	PO	TPPI	NTPPI	EQ, NTPP, PO, TPP, TPPI
TPP	PO	NTPPI	TPPI	EQ, NTPPI, PO, TPP, TPPI
TPP	PO	NTPPI	NTPPI	$\mathcal{B}_{RCC} \setminus \{DC\}$
TPP	EQ	EQ	TPPI	EQ, TPP
TPP	EQ	TPPI	TPPI	EQ, PO, TPP, TPPI
TPP	EQ	TPPI	NTPPI	EQ, NTPP, PO, TPP, TPPI
TPP	EQ	NTPPI	NTPPI	$\mathcal{B}_{RCC} \setminus \{DC\}$
TPP	TPP	PO	TPPI	PO, TPP
TPP	TPP	PO	NTPPI	NTPP, PO, TPP
TPP	TPP	EQ	TPPI	EQ, TPP
TPP	TPP	EQ	NTPPI	EQ, NTPP, TPP
TPP	TPP	TPP	TPPI	TPP
TPP	TPP	TPP	NTPPI	NTPP, TPP
TPP	TPP	TPPI	TPPI	EQ, PO, TPP, TPPI
TPP	TPP	TPPI	NTPPI	EQ, NTPP, PO, TPP, TPPI
TPP	TPP	NTPPI	NTPPI	$\mathcal{B}_{RCC} \setminus \{DC\}$
TPP	NTPP	PO	NTPPI	EC, NTPP, PO, TPP
TPP	NTPP	EQ	NTPPI	EQ, NTPP, TPP
TPP	NTPP	TPP	NTPPI	NTPP, TPP
TPP	NTPP	TPPI	NTPPI	EC, EQ, NTPP, PO, TPP, TPPI
TPP	NTPP	NTPPI	NTPPI	$\mathcal{B}_{RCC} \setminus \{DC\}$

$R(O_1^*, O_2^*)$	$R(O_1^-, O_2^-)$	$R(O_1^*, O_2^-)$	$R(O_1^-, O_2^*)$	$R(O_1^*, O_2^*)$
TPP	TPPI	TPPI	TPPI	EQ, PO, TPP, TPPI
TPP	TPPI	TPPI	NTPPI	EQ, NTPP, PO, TPP, TPPI
TPP	TPPI	NTPPI	TPPI	EQ, NTPPI, PO, TPP, TPPI
TPP	TPPI	NTPPI	NTPPI	$\mathcal{B}_{RCC} \setminus \{DC\}$
TPP	NTPPI	NTPPI	TPPI	EC, EQ, NTPPI, PO, TPP, TPPI
TPP	NTPPI	NTPPI	NTPPI	$\mathcal{B}_{RCC} \setminus \{DC\}$
NTPP	DC	DC	NTPPI	DC, EC, NTPP, PO, TPP
NTPP	DC	EC	NTPPI	DC, EC, NTPP, PO, TPP
NTPP	DC	PO	NTPPI	DC, EC, NTPP, PO, TPP
NTPP	DC	TPPI	NTPPI	$\mathcal{B}_{RCC} \setminus \{NTPPI\}$
NTPP	DC	NTPPI	NTPPI	\mathcal{B}_{RCC}
NTPP	EC	EC	NTPPI	EC, NTPP, PO, TPP
NTPP	EC	PO	NTPPI	EC, NTPP, PO, TPP
NTPP	EC	TPPI	NTPPI	EC, EQ, NTPP, PO, TPP, TPPI
NTPP	EC	NTPPI	NTPPI	$\mathcal{B}_{RCC} \setminus \{DC\}$
NTPP	PO	PO	NTPPI	NTPP, PO, TPP
NTPP	PO	TPPI	NTPPI	EQ, NTPP, PO, TPP, TPPI
NTPP	PO	NTPPI	NTPPI	$\mathcal{B}_{RCC} \setminus \{DC\}$
NTPP	EQ	EQ	NTPPI	EQ, NTPP, TPP
NTPP	EQ	TPPI	NTPPI	EQ, NTPP, PO, TPP, TPPI
NTPP	EQ	NTPPI	NTPPI	$\mathcal{B}_{RCC} \setminus \{DC\}$
NTPP	TPP	PO	NTPPI	NTPP, PO, TPP
NTPP	TPP	EQ	NTPPI	EQ, NTPP, TPP
NTPP	TPP	TPP	NTPPI	NTPP, TPP
NTPP	TPP	TPPI	NTPPI	EQ, NTPP, PO, TPP, TPPI
NTPP	TPP	NTPPI	NTPPI	$\mathcal{B}_{RCC} \setminus \{DC\}$
NTPP	NTPP	PO	NTPPI	EC, NTPP, PO, TPP
NTPP	NTPP	EQ	NTPPI	EQ, NTPP, TPP
NTPP	NTPP	TPP	NTPPI	NTPP, TPP
NTPP	NTPP	NTPP	NTPPI	NTPP
NTPP	NTPP	TPPI	NTPPI	EC, EQ, NTPP, PO, TPP, TPPI
NTPP	NTPP	NTPPI	NTPPI	$\mathcal{B}_{RCC} \setminus \{DC\}$
NTPP	TPPI	TPPI	NTPPI	EQ, NTPP, PO, TPP, TPPI
NTPP	TPPI	NTPPI	NTPPI	$\mathcal{B}_{RCC} \setminus \{DC\}$
NTPP	NTPPI	NTPPI	NTPPI	$\mathcal{B}_{RCC} \setminus \{DC\}$
TPPI	DC	TPPI	DC	DC, EC, PO, TPPI
TPPI	DC	TPPI	EC	DC, EC, PO, TPPI
TPPI	DC	TPPI	PO	DC, EC, PO, TPPI
TPPI	DC	TPPI	TPPI	DC, EC, EQ, PO, TPP, TPPI
TPPI	DC	TPPI	NTPPI	$\mathcal{B}_{RCC} \setminus \{NTPPI\}$
TPPI	DC	NTPPI	DC	DC, EC, NTPPI, PO, TPP
TPPI	DC	NTPPI	EC	DC, EC, NTPPI, PO, TPP
TPPI	DC	NTPPI	PO	DC, EC, NTPPI, PO, TPP
TPPI	DC	NTPPI	TPPI	$\mathcal{B}_{RCC} \setminus \{NTPP\}$
TPPI	DC	NTPPI	NTPPI	\mathcal{B}_{RCC}
TPPI	EC	TPPI	EC	EC, PO, TPPI
TPPI	EC	TPPI	PO	EC, PO, TPPI
TPPI	EC	TPPI	TPPI	EC, EQ, PO, TPP, TPPI
TPPI	EC	TPPI	NTPPI	EC, EQ, NTPP, PO, TPP, TPPI
TPPI	EC	NTPPI	EC	EC, NTPPI, PO, TPPI
TPPI	EC	NTPPI	PO	EC, NTPPI, PO, TPPI
TPPI	EC	NTPPI	TPPI	EC, EQ, NTPPI, PO, TPP, TPPI
TPPI	EC	NTPPI	NTPPI	$\mathcal{B}_{RCC} \setminus \{DC\}$
TPPI	PO	TPPI	PO	PO, TPPI
TPPI	PO	TPPI	TPPI	EQ, PO, TPP, TPPI
TPPI	PO	TPPI	NTPPI	EQ, NTPP, PO, TPP, TPPI
TPPI	PO	NTPPI	PO	NTPPI, PO, TPPI
TPPI	PO	NTPPI	TPPI	EQ, NTPPI, PO, TPP, TPPI
TPPI	PO	NTPPI	NTPPI	$\mathcal{B}_{RCC} \setminus \{DC\}$
TPPI	EQ	TPPI	EQ	EQ, TPPI
TPPI	EQ	TPPI	TPPI	EQ, PO, TPP, TPPI
TPPI	EQ	NTPPI	TPPI	EQ, NTPPI, PO, TPP, TPPI
TPPI	EQ	NTPPI	NTPPI	$\mathcal{B}_{RCC} \setminus \{DC\}$
TPPI	TPP	TPPI	TPPI	EQ, PO, TPP, TPPI
TPPI	TPP	TPPI	NTPPI	EQ, NTPP, PO, TPP, TPPI
TPPI	TPP	NTPPI	TPPI	EQ, NTPPI, PO, TPP, TPPI
TPPI	TPP	NTPPI	NTPPI	$\mathcal{B}_{RCC} \setminus \{DC\}$
TPPI	NTPP	TPPI	NTPPI	EC, EQ, NTPP, PO, TPP, TPPI
TPPI	NTPP	NTPPI	NTPPI	$\mathcal{B}_{RCC} \setminus \{DC\}$

$R(O_1^*, O_2^*)$	$R(O_1^-, O_2^-)$	$R(O_1^*, O_2^-)$	$R(O_1^-, O_2^*)$	$R(O_1^*, O_2^*)$
TPPI	TPPI	TPPI	PO	PO, TPPI
TPPI	TPPI	TPPI	EQ	EQ, TPPI
TPPI	TPPI	TPPI	TPP	TPPI
TPPI	TPPI	TPPI	TPPI	EQ, PO, TPP, TPPI
TPPI	TPPI	NTPPI	PO	NTPPI, PO, TPPI
TPPI	TPPI	NTPPI	EQ	EQ, NTPPI, TPPI
TPPI	TPPI	NTPPI	TPP	NTPPI, TPPI
TPPI	TPPI	NTPPI	TPPI	EQ, NTPPI, PO, TPP, TPPI
TPPI	TPPI	NTPPI	NTPPI	$\mathcal{B}_{RCC} \setminus \{DC\}$
TPPI	NTPPI	NTPPI	PO	EC, NTPPI, PO, TPPI
TPPI	NTPPI	NTPPI	EQ	EQ, NTPPI, TPPI
TPPI	NTPPI	NTPPI	TPP	NTPPI, TPPI
TPPI	NTPPI	NTPPI	TPPI	EC, EQ, NTPPI, PO, TPP, TPPI
TPPI	NTPPI	NTPPI	NTPPI	$\mathcal{B}_{RCC} \setminus \{DC\}$
NTPPI	DC	NTPPI	DC	DC, EC, NTPPI, PO, TPPI
NTPPI	DC	NTPPI	EC	DC, EC, NTPPI, PO, TPPI
NTPPI	DC	NTPPI	PO	DC, EC, NTPPI, PO, TPPI
NTPPI	DC	NTPPI	TPPI	$\mathcal{B}_{RCC} \setminus \{NTPPI\}$
NTPPI	DC	NTPPI	NTPPI	\mathcal{B}_{RCC}
NTPPI	EC	NTPPI	EC	EC, NTPPI, PO, TPPI
NTPPI	EC	NTPPI	PO	EC, NTPPI, PO, TPPI
NTPPI	EC	NTPPI	TPPI	EC, EQ, NTPPI, PO, TPP, TPPI
NTPPI	EC	NTPPI	NTPPI	$\mathcal{B}_{RCC} \setminus \{DC\}$
NTPPI	PO	NTPPI	PO	NTPPI, PO, TPPI
NTPPI	PO	NTPPI	TPPI	EQ, NTPPI, PO, TPP, TPPI
NTPPI	PO	NTPPI	NTPPI	$\mathcal{B}_{RCC} \setminus \{DC\}$
NTPPI	EQ	NTPPI	EQ	EQ, NTPPI, TPPI
NTPPI	EQ	NTPPI	TPPI	EQ, NTPPI, PO, TPP, TPPI
NTPPI	EQ	NTPPI	NTPPI	$\mathcal{B}_{RCC} \setminus \{DC\}$
NTPPI	TPP	NTPPI	TPPI	EQ, NTPPI, PO, TPP, TPPI
NTPPI	TPP	NTPPI	NTPPI	$\mathcal{B}_{RCC} \setminus \{DC\}$
NTPPI	NTPP	NTPPI	NTPPI	$\mathcal{B}_{RCC} \setminus \{DC\}$
NTPPI	TPPI	NTPPI	PO	NTPPI, PO, TPPI
NTPPI	TPPI	NTPPI	EQ	EQ, NTPPI, TPPI
NTPPI	TPPI	NTPPI	TPP	NTPPI, TPPI
NTPPI	TPPI	NTPPI	TPPI	EQ, NTPPI, PO, TPP, TPPI
NTPPI	TPPI	NTPPI	NTPPI	$\mathcal{B}_{RCC} \setminus \{DC\}$
NTPPI	NTPPI	NTPPI	PO	EC, NTPPI, PO, TPPI
NTPPI	NTPPI	NTPPI	EQ	EQ, NTPPI, TPPI
NTPPI	NTPPI	NTPPI	TPP	NTPPI, TPPI
NTPPI	NTPPI	NTPPI	NTPP	NTPPI
NTPPI	NTPPI	NTPPI	TPPI	EC, EQ, NTPPI, PO, TPP, TPPI
NTPPI	NTPPI	NTPPI	NTPPI	$\mathcal{B}_{RCC} \setminus \{DC\}$

5.3 Qualification of Visibility Relations

In this section, the qualification of visibility relations will be analyzed. Given three spatial-region objects, the qualification of the visibility crisp disjunctive relation between them is defined, from Definition 12, as:

Definition 16 (Visibility crisp relation between imprecise objects). *Let O_1^* , O_2^* and O_3^* be three imprecise spatial-region objects, the disjunctive visibility relation that describes the possible relations holding between any precise instance of O_1^* , O_2^* , and O_3^* , called visibility crisp relation between imprecise objects, is defined as:*

$$R_{Vis}(O_1^*, O_2^*, O_3^*) \triangleq \left\{ R \in U_{Vis} \mid \exists O_1^\circ, \exists O_2^\circ, \exists O_3^\circ \text{ s.t. } R(O_1^\circ, O_2^\circ, O_3^\circ) \right\}$$

Table 5.2: Qualification of RCC-8 relations – $R_{RCC}(O_1, O_2^+)$, $R_{RCC}(O_1, O_2^*)$, $R_{RCC}(O_1^+, O_2^+)$, and $R_{RCC}(O_1^+, O_2^*)$.

Pri	Ref	$R(O_1^*, O_2^*)$	$R(O_1^*, O_2^-)$	$R(O_1^*, O_2^*)$
O_1	O_2^+	DC	–	DC
O_1	O_2^+	EC	–	DC, EC
O_1	O_2^+	PO	–	$DC, EC, PO, NTPPI, TPPI$
O_1	O_2^+	EQ	–	$EQ, NTPPI, TPPI$
O_1	O_2^+	TPP	–	$\mathcal{B}_{RCC} \setminus \{NTPPI\}$
O_1	O_2^+	$NTPP$	–	\mathcal{B}_{RCC}
O_1	O_2^+	$TPPI$	–	$TPPI, NTPPI$
O_1	O_2^+	$NTPPI$	–	$NTPPI$
O_1	O_2^*	DC	\mathcal{B}_{RCC}	DC
O_1	O_2^*	EC	\mathcal{B}_{RCC}	DC, EC
O_1	O_2^*	PO	\mathcal{B}_{RCC}	$DC, EC, PO, NTPPI, TPPI$
O_1	O_2^*	EQ	\mathcal{B}_{RCC}	$EQ, NTPPI, TPPI$
O_1	O_2^*	TPP	\mathcal{B}_{RCC}	$\mathcal{B}_{RCC} \setminus \{NTPPI\}$
O_1	O_2^*	$NTPP$	\mathcal{B}_{RCC}	\mathcal{B}_{RCC}
O_1	O_2^*	$TPPI$	\mathcal{B}_{RCC}	$TPPI, NTPPI$
O_1	O_2^*	$NTPPI$	\mathcal{B}_{RCC}	$NTPPI$
O_1^+	O_2^+	DC	–	DC
O_1^+	O_2^+	EC	–	DC, EC
O_1^+	O_2^+	$\mathcal{B}_{RCC} \setminus \{DC, EC\}$	–	\mathcal{B}_{RCC}
O_1^+	O_2^*	DC	DC	DC
O_1^+	O_2^*	EC	DC	DC, EC
O_1^+	O_2^*	EC	EC	EC
O_1^+	O_2^*	PO	DC	DC, EC, PO
O_1^+	O_2^*	PO	EC	EC, PO
O_1^+	O_2^*	PO	PO	PO
O_1^+	O_2^*	PO	$TPPI$	$PO, TPPI$
O_1^+	O_2^*	PO	$NTPPI$	$EC, PO, TPPI, NTPPI$
O_1^+	O_2^*	EQ	EQ	EQ
O_1^+	O_2^*	EQ	$TPPI$	$EQ, TPPI$
O_1^+	O_2^*	EQ	$NTPPI$	$EQ, TPPI, NTPPI$
O_1^+	O_2^*	TPP	DC	DC, EC, PO, TPP
O_1^+	O_2^*	TPP	EC	EC, PO, TPP
O_1^+	O_2^*	TPP	PO	PO, TPP
O_1^+	O_2^*	TPP	EQ	EQ, TPP
O_1^+	O_2^*	TPP	TPP	TPP
O_1^+	O_2^*	TPP	$TPPI$	$EQ, PO, TPP, TPPI$
O_1^+	O_2^*	TPP	$NTPPI$	$EC, PO, EQ, TPP, TPPI, NTPPI$
O_1^+	O_2^*	$NTPP$	DC	$DC, EC, PO, TPP, NTPP$
O_1^+	O_2^*	$NTPP$	EC	$EC, PO, TPP, NTPP$
O_1^+	O_2^*	$NTPP$	PO	$PO, TPP, NTPP$
O_1^+	O_2^*	$NTPP$	EQ	$EQ, TPP, NTPP$
O_1^+	O_2^*	$NTPP$	TPP	$TPP, NTPP$
O_1^+	O_2^*	$NTPP$	$NTPP$	$NTPP$
O_1^+	O_2^*	$NTPP$	$TPPI$	$EQ, PO, TPP, NTPP, TPPI$
O_1^+	O_2^*	$NTPP$	$NTPPI$	$\mathcal{B}_{RCC} \setminus \{DC\}$
O_1^+	O_2^*	$TPPI$	$TPPI$	$TPPI$
O_1^+	O_2^*	$TPPI$	$NTPPI$	$TPPI, NTPPI$
O_1^+	O_2^*	$NTPPI$	$NTPPI$	$NTPPI$

As discussed for topology, a computational procedure to identify $R_{Vis}(O_1^*, O_2^*, O_3^*)$ is required. However, differently from the previous case, it is not possible to exploit the properties of the egg-yolk approach to derive the visibility crisp relation, since the egg-yolk approach is grounded on the topological concept of *containment* and it does not yield any information at the visibility level. Rather, in this case the properties of the quantifications analyzed in Chapter 4 can be used as a basis for the computational procedure. Indeed, from Eq. 4.33, if for a certain R there exist O_1^o , O_2^o , and O_3^o such that $R(O_1^o, O_2^o, O_3^o)$, also $O_1^o \subseteq A_R^+(O_2^*, O_3^*)$ holds true. Hence, it is possible to test which maximal quantifications are crossed by the the egg and the yolk of O_1^* , and from that the visibility crisp relation between imprecise objects can be computed.

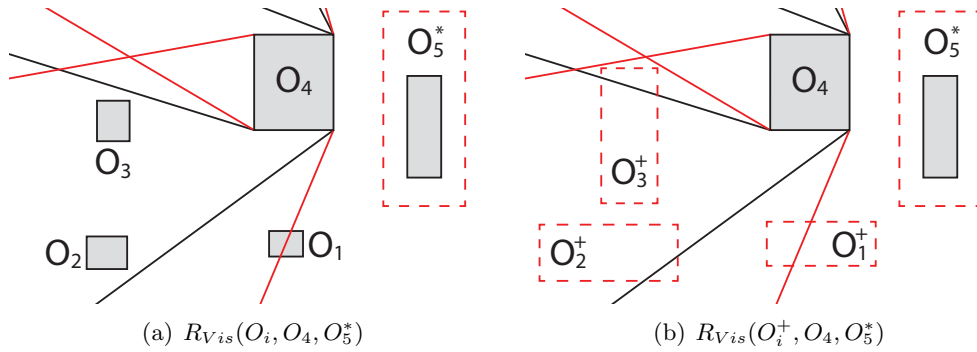


Figure 5.2: Qualification of visibility relations between imprecise objects.

For instance, Fig. 5.2 shows two examples for which one the reference objects is imprecise and the other is precise (i.e., the obstacle object is precise and the viewer is imprecise), and for which the primary object is either precise or imprecise with empty yolk. In Fig. 5.2(a) three distinct precise primary objects are depicted. The object O_1 crosses two different maximal quantifications: $A_V^+(O_4, O_5^*)$ and $A_{PV_L}^+(O_4, O_5^*)$ (cf. Fig. 4.13). However, the object is fully contained in the former quantification, while it just intersects the latter. It is possible to identify a precise instance of O_5^* (namely $O_5^o = O_5^-$) such that $V(O_1, O_4, O_5^o)$ holds, and a different precise instance ($O_5^o = O_5^\bullet$) such that $\{V:PV_L\}(O_1, O_4, O_5^o)$ is satisfied. In contrast, there does not exist a precise instance of O_5^* such that only $PV_L(O_1, O_4, O_5^o)$ holds. Hence, $R_{Vis}(O_1, O_4, O_5^*) = \{V, V:PV_L\}$ that is equivalent to the δ^1 of the quantifications crossed by O_1 , excluding those relations for which O_1 is not fully contained in the maximal quantification. Furthermore, the other two objects depicted in Fig. 5.2(a) are fully contained in $A_{PV_L}^+(O_4, O_5^*)$ and the only accepted relation is PV_L . The result can be generalized as in Proposition 5.3.1.

¹The function $\delta(R_1, \dots, R_n)$ has been defined in Section 2.2.3.2 (p. 33). Given a set of single-tile relations, it yields all the single and multi-tile relations that can be obtained by combining the input set. Even though the function has been defined for CDC relations, it applies also to visibility ones.

Proposition 5.3.1. *Let O_1 be a precise object, and let O_2^*, O_3^* be two imprecise spatial-region objects, the visibility crisp relation $R_{Vis}(O_1, O_2^*, O_3^*)$ satisfies the following constraints:*

- (i) $R_{Vis}(O_1, O_2^*, O_3^*) \subseteq \delta(T)$ with $T = \left\{ R \mid R \in \mathcal{R}_{Vis}^{ST} \wedge O_1 \cap A_R^+(O_2^*, O_3^*) \neq \emptyset \right\}$
- (ii) $\forall R : R \in R_{Vis}(O_1, O_2^*, O_3^*) \rightarrow O_1 \subseteq \left(\bigcup_{T \in \Gamma(R)} A_T^+(O_2^*, O_3^*) \right)$

In Fig. 5.2(b) the case where the primary objects are imprecise with empty yolk is shown. Differently from the example above, the primary object is not precise, thus all its precise instances have to be considered. For instance, the object O_1^+ crosses two distinct maximal quantifications— $A_V^+(O_4, O_5^*)$ and $A_{PV_L}^+(O_4, O_5^*)$ —and three precise instance of O_1^+ can be identified such that $R(O_1^o, O_4, O_5^*) = \delta(V, PV_L)$. Hence, $R_{Vis}(O_1^+, O_4, O_5^*) = \delta(V, PV_L)$. Similar results can be achieved for the objects O_2^+ and O_3^+ : $R_{Vis}(O_2^+, O_4, O_5^*) = \delta(V, PV_L)$, $R_{Vis}(O_3^+, O_4, O_5^*) = \delta(OC, PV_L)$. A generalization of this result is done in Proposition 5.3.2.

Proposition 5.3.2. *Let O_1^+ be an imprecise object with empty yolk, and let O_2^*, O_3^* be two imprecise spatial-region objects, the visibility crisp relation $R_{Vis}(O_1^+, O_2^*, O_3^*)$ between them is:*

$$R_{Vis}(O_1^+, O_2^*, O_3^*) = \delta(T) \text{ with } T = \left\{ R \mid R \in \mathcal{R}_{Vis}^{ST} \wedge O_1^+ \cap A_R^+(O_2^*, O_3^*) \neq \emptyset \right\}$$

Finally, the computation of the disjunctive crisp relation for the case if the primary object is imprecise with non-empty yolk can be done by combining the results of Proposition 5.3.1 and Proposition 5.3.2. In particular, the quantifications crossed by the egg of the primary object define the superset of the accepted relations, while the quantifications crossed by the yolk are used to refine the set. Indeed, any precise instance of the primary object contains the yolk, and the relation

satisfied by the precise instance has to contain the tiles of the relation satisfied by the yolk. Of course, any possible relation that the yolk can satisfy has to be considered. As an example, considering the regions in Fig. 5.3, the visibility crisp relations between imprecise objects are: $R_{Vis}(O_1^*, O_4, O_5^*) = \{V, V:PV_L\}$, $R_{Vis}(O_2^*, O_4, O_5^*) = \{V:PV_L\}$, and $R_{Vis}(O_3^*, O_4, O_5^*) = \{OC:PV_L\}$. The result is generalized as in Proposition 5.3.3.

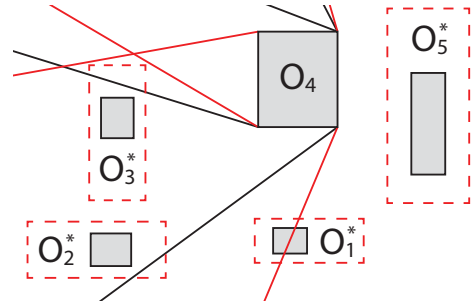


Figure 5.3: $R_{Vis}(O_i^*, O_4, O_5^*)$.

Proposition 5.3.3. *Let O_1^*, O_2^*, O_3^* be three imprecise spatial-region objects, and let R_E and R_Y be defined respectively as $R_E \triangleq R_{Vis}(O_1^+, O_2^*, O_3^*)$ and $R_Y \triangleq R_{Vis}(O_1^-, O_2^*, O_3^*)$, the visibility crisp relation $R_{Vis}(O_1^*, O_2^*, O_3^*)$ satisfies the following constraints:*

- (i) $R_{Vis}(O_1^*, O_2^*, O_3^*) \subseteq R_E$
- (ii) $\forall R_1 \in R_{Vis}(O_1^*, O_2^*, O_3^*), \exists R_2 \in R_Y : \Gamma(R_1) \supseteq \Gamma(R_2)$

Hence, Proposition 5.3.1, Proposition 5.3.2, and Proposition 5.3.3 provide a procedure to compute the visibility crisp relation between imprecise objects by considering the imprecision of the primary spatial-region object.

5.4 Qualification of Cardinal Direction Relations

This section aims at the definition of algorithms for qualifying cardinal direction relations between imprecise objects. At first, the strategy developed in the previous section for the visibility calculus is directly adapted to deal with CDC relations; the same approach can be adopted for qualifying relations defined in other projective calculi. However, the structure of the CDC calculus can be exploited in order to reduce the complexity of the procedures, as it will be discussed from Section 5.4.1.

Given two imprecise spatial-region objects, the qualification of the CDC crisp relation between them is defined, from Definition 12, as:

Definition 17 (CDC crisp relation between imprecise objects).

Let O_1^, O_2^* be two imprecise spatial-region objects, the disjunctive cardinal direction relation that describes the possible relations holding between any precise instance of O_1^* and any precise instance of O_2^* , called CDC crisp relation between imprecise objects, is defined as:*

$$R_{CDC}(O_1^*, O_2^*) \triangleq \left\{ R \in U_{CDC} \mid \exists O_1^\circ, \exists O_2^\circ \text{ s.t. } R(O_1^\circ, O_2^\circ) \right\}$$

At first, the crisp relation can be computed by checking which quantifications the primary object lies along, as it has been shown for visibility in the previous section.

As an example, considering the spatial-region objects depicted in Fig. 5.4(a), O_1 crosses both $A_{NE}^+(O_2^*)$ and $A_E^+(O_2^*)$, and $\delta(NE, E) = \{NE, E, NE:E\}$. However, as the picture shows there exist two precise instances of O_2^* such that $E(O_1, O_2^\circ)$ and $NE:E(O_1, O_2^\circ)$, but it does not exist a precise instance for which $NE(O_1, O_2^\circ)$ is satisfied. Hence, $R_{CDC}(O_1, O_2^*) = \{E, NE:E\} \subset \delta(NE, E)$. Instead, considering the two objects depicted in Fig. 5.4(b), the object O_1^* crosses both $A_{NE}^+(O_2)$ and $A_E^+(O_2)$. As the pictures shows, it is possible to identify three precise instance of O_1^+ such that $NE(O_1^\circ, O_2)$,

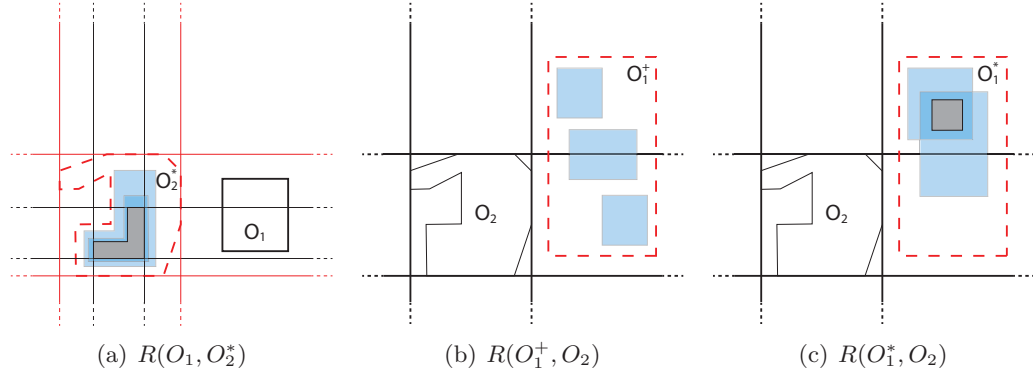


Figure 5.4: Qualification of CDC relations between imprecise objects.

$E(O_1^o, O_2)$, and $NE:E(O_1^o, O_2)$; hence, $R_{CDC}(O_1^+, O_2) = \{NE, E, NE:E\} = \delta(NE, E)$, that corresponds to the δ of the tiles crossed by O_1^\bullet . Finally, Fig 5.4(c) shows an example in which the reference object is precise, while the primary object is imprecise with non-empty yolk. In this case, the relation satisfied by the yolk of the primary object constraints the CDC crisp relation between imprecise objects. It results $R_{CDC}(O_1^*, O_2) = \{NE, E:NE\}$. This results can be generalized for a pair of spatial-region objects in the same way it has been done for the visibility case in the previous section, as Proposition 5.4.1, Proposition 5.4.2 and Proposition 5.4.3 show.

Proposition 5.4.1. *Let O_1 be a precise object and let O_2^* be an imprecise spatial-region object, the CDC crisp relation $R_{CDC}(O_1, O_2^*)$ satisfies the following constraints:*

- (i) $R_{CDC}(O_1, O_2^*) \subseteq \delta(T)$ with $T = \left\{ R \mid R \in \mathcal{R}_{CDC}^{ST} \wedge O_1 \cap A_R^+(O_2^*) \neq \emptyset \right\}$
- (ii) $\forall R : R \in R_{CDC}(O_1, O_2^*) \rightarrow O_1 \subseteq \left(\bigcup_{T \in \Gamma(R)} A_T^+(O_2^*) \right)$

Proposition 5.4.2. *Let O_1^+ be an imprecise object with empty yolk, and let O_2^* be an imprecise spatial-region object, the CDC crisp relation $R_{CDC}(O_1^+, O_2^*)$ is:*

$$R_{CDC}(O_1^+, O_2^*) = \delta(T) \text{ with } T = \left\{ R \mid R \in \mathcal{R}_{CDC}^{ST} \wedge O_1^\bullet \cap A_R^+(O_2^*) \neq \emptyset \right\}$$

Proposition 5.4.3. *Let O_1^* and O_2^* be two imprecise spatial-region objects, and let R_E and R_Y be two relations defined respectively as $R_E \triangleq R_{CDC}(O_1^+, O_2^*)$ and $R_Y \triangleq$*

$R_{CDC}(O_1^-, O_2^*)$, the CDC crisp relation $R_{CDC}(O_1^*, O_2^*)$ satisfies the following constraints:

- (i) $R_{CDC}(O_1^*, O_2^*) \subseteq R_E$
- (ii) $\forall R_1 \in R_{CDC}(O_1^*, O_2^*), \exists R_2 \in R_Y : \Gamma(R_1) \supseteq \Gamma(R_2)$

Even though this procedure already provides a computational approach to retrieve the CDC crisp relation between imprecise objects, the geometrical properties of the CDC frame of reference can be exploited in order to reduce the computational time complexity of the procedure. Indeed, as previously shown in Section 4.7.1, the computation of the quantification of a cardinal direction relation whose reference object is precise requires linear time, while if the object is imprecise the time required increases exponentially with the number of its vertices. Thus, a theory to compute the relation by calculating only the quantifications over precise objects will be developed. At first, the cases if one of the objects is either precise or imprecise with empty yolk will be investigated. The results will then be propagated to the general cases of imprecise spatial-region objects.

5.4.1 Precise Reference Object

If both objects are precise, the qualitative cardinal direction relation holding between them ($R_{CDC}(O_1, O_2)$) is computed by checking the relations' constraints defined in Eq. 2.3-2.11 (Skiadopoulos & Koubarakis, 2004). Alternatively, the function *Quantify_C* developed in the previous chapter can be exploited, as Equation 5.1 shows.

$$R_{CDC}(O_1, O_2) = \left\{ T_1 : \dots : T_n \mid \forall i = 1 \dots n, T_i \in \mathcal{R}_{CDC}^{ST} \wedge O_1 \cap A_{T_i}^+(O_2) \neq \emptyset \right\} \quad (5.1)$$

If the reference object is precise, and the primary one is imprecise either with or without empty yolk, the CDC crisp disjunctive relation between them can be computed respectively as in Theorem 5.4.1 and Theorem 5.4.2.

Theorem 5.4.1. *Let O_1^+ be an imprecise spatial-region object with empty yolk, and let O_2 be a precise object, then:*

$$R_{CDC}(O_1^+, O_2) = \delta\left(\Gamma(R_{CDC}(O_1^\bullet, O_2))\right) \quad (5.2)$$

Proof. $R_{CDC}(O_1^\bullet, O_2) = \{T_1 : \dots : T_n\}$ is a single or multi-tile relation, with $n \geq 1$. The precise object O_1^\bullet can be partitioned in n subregions based on the CDC tiles: $O_1^\bullet = \bigcup_{i=1, \dots, n} o_i, \forall i \neq j \rightarrow o_i \cap o_j = \emptyset \wedge o_i \cap T_j = \emptyset$, and $\forall i \leq n \rightarrow o_i \cap T_i = o_i$. In other words, any o_i represents the subregion of O_1^\bullet that is contained in the tile T_i .

For any relation $R \in \delta(\Gamma(R_{CDC}(O_1^\bullet, O_2))) \Rightarrow R \in \delta(T_1, \dots, T_n)$. Furthermore $R = \{T_{R1} : \dots : T_{Rk}\}$ with $k \leq n; \forall j \leq k \rightarrow T_{Rj} \in \{T_1, \dots, T_n\}$. For any relation $R \in \delta(T_1, \dots, T_n)$, an object O_1°

precise instance of O_1^+ that satisfies the relation $R(O_1^\circ, O_2)$ is build as $O_1^\circ = \bigcup_{j=R1, \dots, Rk} o_j$. Hence Eq. 5.2 holds true. \square

Theorem 5.4.2. *Let O_1^* be an imprecise spatial-region object, and O_2 be a precise object, then:*

$$R_{CDC}(O_1^*, O_2) = \left\{ R \mid R \in R_{CDC}(O_1^+, O_2) \wedge \Gamma(R) \supseteq \Gamma(R_{CDC}(O_1^-, O_2)) \right\} \quad (5.3)$$

Proof. The first condition of the equation directly derives from Theorem 5.4.1 by considering the constraint $O_1^\circ \subseteq O_1^+$. Furthermore, the second condition is trivially proven by observing that $O_1^\circ \supseteq O_1^-$, hence being $R_{CDC}(O_1^-, O_2) = \{T_1 : \dots : T_n\}$, $O_1^\circ \cap T_i \neq \emptyset, \forall i = 1, \dots, n$. Hence, $\Gamma(R) \supseteq \Gamma(R_{CDC}(O_1^-, O_2))$. \square

Considering again the objects depicted in Fig. 5.4(b), the relation $R_{CDC}(O_1^\bullet, O_2)$ is $NE:E$, and from Theorem 5.4.1 it results that $R_{CDC}(O_1^+, O_2) = \{E, NE, NE:E\}$. This result is equivalent to the one obtained from the constraints in Proposition 5.4.2. Similarly, the result of Theorem 5.4.2 for the objects in Fig. 5.4(c) is $R_{CDC}(O_1^*, O_2) = \{NE, E:NE\}$ that is equivalent to Proposition 5.4.3 (cf. Section 5.4).

5.4.2 Imprecise Reference Object

In the following, a theory to compute the disjunctive crisp cardinal direction relation in the case the reference object is imprecise is developed. It exploits the relations between the eggs and the yolks of the two objects interpreted as precise objects. Some operators required for the definition of the theory will be introduced beforehand.

5.4.2.1 Side, Angular and Wing Operators

At first, a function *Side* is defined that, given a single-tile relation $R^{ST} \in \mathcal{R}_{CDC}^{ST}$, returns the tiles of the R^{ST} -most¹ rectangular relation of the multi-tile relation formed by all the CDC tiles.

Definition 18 (*Side operator*).

Let $R^{ST} \in \{N, E, S, W\}$, the function $Side(R^{ST})$ yields:

$$Side(R^{ST}) \triangleq \Gamma(Most(R^{ST}, \{N:NE:E:SE:S:SW:W:NW:B\}))$$

The values returned by the *Side* operator $Side(R^{ST})$ are reported in Tab. 5.3. The conditions to check whether two precise objects O_1 and O_2 are in a relation belonging to $\delta(Side(R^{ST}))$ can be easily derived from Eq. 2.3-2.11, as Lemma 5.4.3 shows.

¹The R^{ST} -most concept and the function $Most(R^{ST}, R^{REC})$ have been defined in Sec.2.2.3.2.

Table 5.3: Side(R^{ST}) operator.

R^{ST}	N	E	S	W
Side(R^{ST})	N,NE,NW	NE,E,SE	SE,S,SW	SW,W,NW

Lemma 5.4.3. *Let O_1 and O_2 be two precise objects, the following conditions hold:*

$$\begin{aligned}
R(O_1, O_2) \in \delta(\text{Side}(N)) & \quad \text{iff} \quad \underline{Y}(O_1) \geq \overline{Y}(O_2) \\
R(O_1, O_2) \in \delta(\text{Side}(E)) & \quad \text{iff} \quad \underline{X}(O_1) \geq \overline{X}(O_2) \\
R(O_1, O_2) \in \delta(\text{Side}(S)) & \quad \text{iff} \quad \overline{Y}(O_1) \leq \underline{Y}(O_2) \\
R(O_1, O_2) \in \delta(\text{Side}(W)) & \quad \text{iff} \quad \overline{X}(O_1) \leq \underline{X}(O_2)
\end{aligned}$$

Besides, two auxiliary functions (i.e., *Angular* and *Wing*) that yield specific tiles of a given CDC multi-tile relation are defined in Definition 19. As an example, given $R^{MT} = \{N:NE:NW:B\}$, it results that: $\angle(R^{MT}) = \{NE, NW\}$ and $\square(R^{MT}) = \{N\}$.

Definition 19 (*Angular and Wing operators*).

Let $R^{MT} = \{T_1 : \dots : T_k\}$ be a CDC multi-tile relation, the operator *Angular*, denoted as $\angle(R^{MT})$, yields the tiles of R^{MT} that belong to the set $\{NE, SE, SW, NW\}$, while the *Wing* operator, denoted as $\square(R^{MT})$, yields the tiles that belong to $\{N, E, S, W\}$:

$$\begin{aligned}
\angle(R^{MT}) &\triangleq \{T_i \mid T_i \in \Gamma(R^{MT}) \wedge T_i \in \{NE, SE, SW, NW\}\} \\
\square(R^{MT}) &\triangleq \{T_i \mid T_i \in \Gamma(R^{MT}) \wedge T_i \in \{N, S, W, N\}\}
\end{aligned}$$

Finally, if the primary object is precise and the reference one is imprecise with empty yolk, from Definition 17 it holds true that:

$$R_{CDC}(O_1, O_2^+) = \{R \in U_{CDC} \mid \exists O_2^\circ \subseteq O_2^+ \text{ s.t. } R(O_1, O_2^\circ)\}$$

In turns, Lemma 5.4.4 directly follows from the above definition.

Lemma 5.4.4. *Let O_1 be a precise object, and let O_2^+ be an imprecise object with empty yolk, it holds true that:*

$$\forall O_2^\circ \subseteq O_2^+, \exists R \in R_{CDC}(O_1, O_2^+) : R(O_1, O_2^\circ)$$

5.4.2.2 A Theory for the Computation of CDC Crisp Relations

Different cases can occur depending on whether the objects are precise, imprecise with empty yolk and imprecise with non-empty yolk. All the possibilities are analyzed

separately. Some examples of the different combinations of the objects' properties are shown in Fig 5.5.

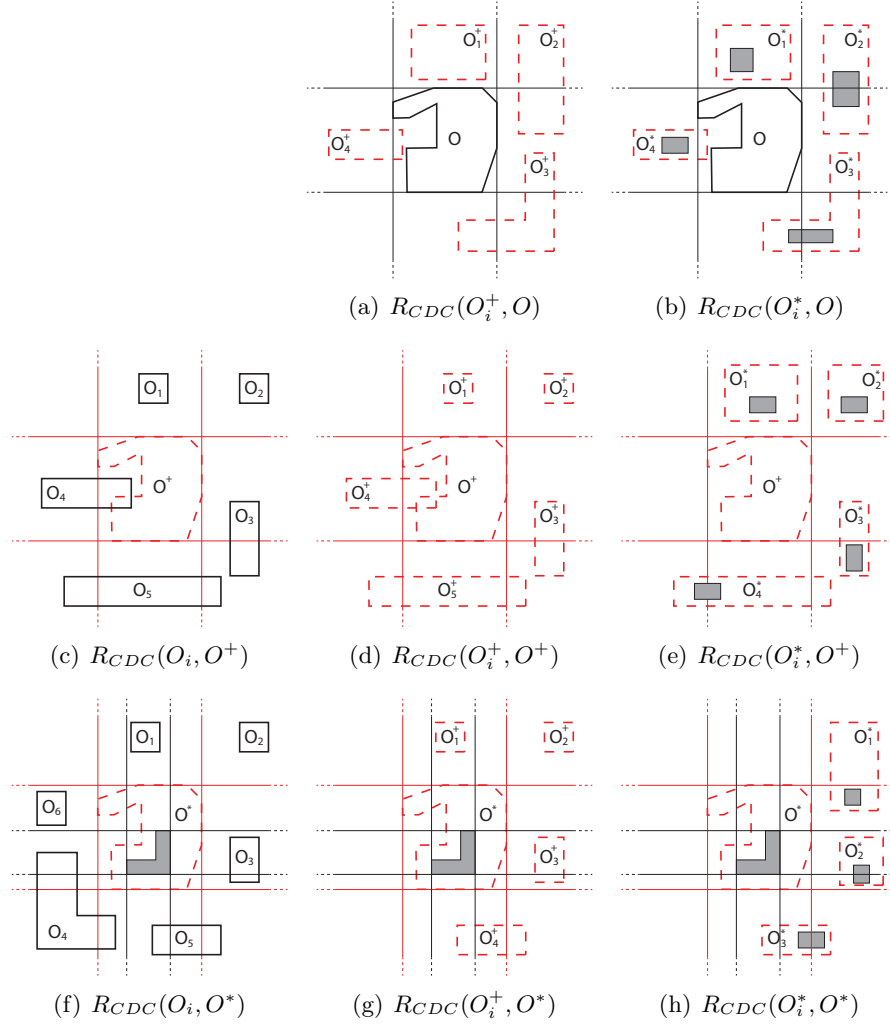


Figure 5.5: Cardinal direction crisp relations between imprecise objects.

The first case analyzed is the one in which the primary object is precise, while the reference one is imprecise with empty yolk. The procedure to identify the relation between them relies only on the computation of $R_{CDC}(O_1, O_2^\bullet)$. If the relation holding between O_1 and O_2^\bullet is a single-tile relation, it is directly possible to compute the CDC crisp relation $R_{CDC}(O_1, O_2^+)$ as it is shown in Theorem 5.4.5. Instead, if $R_{CDC}(O_1, O_2^\bullet)$ is a multi-tile CDC relation, the CDC crisp relation between imprecise objects is computed as Theorem 5.4.6 shows.

Theorem 5.4.5. *Let O_1 be a precise object and let O_2^+ be an imprecise spatial-region object, if $R^{ST} = R_{CDC}(O_1, O_2^\bullet)$ is a single-tile relation then the following conditions hold:*

$$(i) \quad R_{CDC}(O_1, O_2^+) = R^{ST} \quad \text{iff} \quad R^{ST} \in \{NE, SE, SW, NW\} \quad (5.4)$$

$$(ii) \quad R_{CDC}(O_1, O_2^+) = \delta(\text{Side}(R^{ST})) \quad \text{iff} \quad R^{ST} \in \{N, S, E, W\} \quad (5.5)$$

$$(iii) \quad R_{CDC}(O_1, O_2^+) \subseteq U_{CDC} \quad \text{iff} \quad R^{ST} = B \quad (5.6)$$

Proof. (i) At first, the case $R^{ST} = NE(O_1, O_2^\bullet)$ is considered. From Eq. 2.5 and Eq. 3.1, $\forall O_2^\circ \subseteq O_2^+$ it holds:

$$\begin{cases} \overline{X}(O_2^\bullet) \leq \underline{X}(O_1) \\ \overline{Y}(O_2^\bullet) \leq \underline{Y}(O_1) \end{cases} \Rightarrow \begin{cases} \overline{X}(O_2^\circ) \leq \underline{X}(O_1) \\ \overline{Y}(O_2^\circ) \leq \underline{Y}(O_1) \end{cases} \Rightarrow NE(O_1, O_2^\circ) \Rightarrow R_{CDC}(O_1, O_2^+) = NE$$

The proofs of the cases $R^{ST} \in \{SE, SW, NW\}$ follow the same procedure.

(ii) At first, the case $R^{ST} = N(O_1, O_2^\bullet)$ is considered. From Eq. 2.4 and Eq. 3.1, it follows that:

$$\begin{cases} \overline{Y}(O_2^\bullet) \leq \underline{Y}(O_1) \\ \underline{X}(O_2^\bullet) \leq \underline{X}(O_1) \\ \overline{X}(O_2^\bullet) \geq \overline{X}(O_1) \end{cases} \Rightarrow \begin{cases} \overline{Y}(O_2^\circ) \leq \underline{Y}(O_1) \\ \underline{X}(O_2^\circ) \leq \underline{X}(O_1) \\ \overline{X}(O_2^\circ) \leq \overline{X}(O_1) \end{cases} \quad \forall O_2^\circ \subseteq O_2^+$$

From Lemma 5.4.3, $\forall O_2^\circ \subseteq O_2^+, R(O_1, O_2^\circ) \in \delta(\text{Side}(N)) \Rightarrow R(O_1, O_2^+) = \delta(\text{Side}(N))$. The proofs of the cases $R^{ST} \in \{E, S, W\}$ follow the same procedure.

(iii) From Eq. 2.3 and Eq. 3.1, it follows that:

$$\begin{cases} \underline{X}(O_2^\bullet) \leq \underline{X}(O_1) \\ \overline{X}(O_2^\bullet) \geq \overline{X}(O_1) \\ \underline{Y}(O_2^\bullet) \leq \underline{Y}(O_1) \\ \overline{Y}(O_2^\bullet) \geq \overline{Y}(O_1) \end{cases} \Rightarrow \begin{cases} \underline{X}(O_2^\circ) \leq \underline{X}(O_1) \\ \overline{X}(O_2^\circ) \geq \overline{X}(O_1) \\ \underline{Y}(O_2^\circ) \leq \underline{Y}(O_1) \\ \overline{Y}(O_2^\circ) \geq \overline{Y}(O_1) \end{cases} \quad \forall O_2^\circ \subseteq O_2^+$$

Since no constraint can be identified for the bounding box of O_2° , it follows that $\forall O_2^\circ \subseteq O_2^+ \rightarrow R(O_1, O_2^\circ) \in U_{CDC}$ and hence $R(O_1, O_2^+) \subseteq U_{CDC}$. \square

Theorem 5.4.6. *Let O_1 be a precise object, O_2^+ be an imprecise spatial-region object with empty yolk, if $R^{MT}(O_1, O_2^\bullet)$ is a CDC multi-tile relation, the following conditions hold:*

$$\begin{aligned} (i) \quad R_{CDC}(O_1, O_2^+) &\subseteq \begin{cases} \delta\left(\bigcup_{T \in \square(R^{MT})} \text{Side}(T)\right) & \text{iff } B \notin \Gamma(R^{MT}) \\ U_{CDC} & \text{iff } B \in \Gamma(R^{MT}) \end{cases} \\ (ii) \quad R_{CDC}(O_1, O_2^+) &\subseteq \left\{ R \mid \Gamma(R) \supseteq \angle(R^{MT}) \right\} \\ (iii) \quad R_{CDC}(O_1, O_2^+) &\subseteq \left\{ R \mid \forall T \in \square(R^{MT}), \exists T' \in \text{Side}(T) \text{ s.t. } T' \in \Gamma(R) \right\} \end{aligned} \quad (5.7)$$

Proof. (i)-(iii) The proof trivially follows from Theorem 5.4.5.

(ii) From the definition of multi-tile relation, $\forall T_{ANG} \in \angle(R^{MT}), \exists o_1 \subseteq O_1 : T_{ANG}(o_1, O_2^\bullet)$. In turns, from Theorem 5.4.5(i), $\forall O_2^\circ \subseteq O_2^+ \Rightarrow T_{ANG}(o_1, O_2^\circ)$. Hence, for any $O_2^\circ \subseteq O_2^+$ a subregion $o_1 \subseteq O_1$ that is inside the tile T_{ANG} always exists and hence $T_{ANG} \in \Gamma(R)$. \square

As an example, the object O_1 in Fig 5.5(c) satisfies the relation $N(O_1, O^\bullet)$; hence, $R_{CDC}(O_1, O^+)$ is computed as in Theorem 5.4.5 and it results: $R_{CDC}(O_1, O^+) = \delta(N, NE, NW)$. Moreover, O_1 crosses the quantifications $A_N^+(O^+)$, $A_{NE}^+(O^+)$ and $A_{NW}^+(O^+)$, and it is totally contained into all of them; thus from Proposition 5.4.1 it also results that $R_{CDC}(O_1, O^+) = \delta(N, NE, NW)$. Hence, in this case Theorem 5.4.5 is equivalent to Proposition 5.4.1. Furthermore, considering that O_3 satisfies $\{E:SE\}(O_3, O^\bullet)$, from Theorem 5.4.6 it results that: (i) $R_{CDC}(O_3, O^+)$ is contained into $\delta(NE, E, SE)$ and (ii) any relation of $R_{CDC}(O_3, O^+)$ contains the tile SE . The latter result also satisfies the condition (iii) of the theorem since $SE \in Side(E)$. Hence, $R_{CDC}(O_3, O^+) = \{SE, E:SE, NE:E:SE\}$. It is easy to proof that O_3 is fully contained only into the quantifications of those three relations, thus also in this case Proposition 5.4.1 is equivalent to Theorem 5.4.6.

Similarly, the CDC crisp relation between imprecise objects can be computed from the relations satisfied by their eggs and their yolks in the cases if both objects are imprecise with empty yolk (Theorem 5.4.7), the primary object is imprecise and the reference one has empty yolk (Theorem 5.4.8), the reference object is imprecise and the primary one is precise (Theorem 5.4.9), the reference is imprecise and the primary has empty yolk (Theorem 5.4.10), and finally if both objects are imprecise with non-empty yolk (Theorem 5.4.11). The qualifications of the CDC relations between the imprecise objects depicted in Fig. 5.5 are summarized in Table 5.4.

Theorem 5.4.7. *Let O_1^+, O_2^+ be two imprecise spatial-region objects with empty yolk that satisfy $R \triangleq R_{CDC}(O_1^\bullet, O_2^\bullet)$, the following equation is valid:*

$$R_{CDC}(O_1^+, O_2^+) \subseteq \bigcup_{T \in \Gamma(R)} \begin{cases} \delta(T \cup \Gamma(R)) & \text{iff } T \in \{NE, SE, SW, NW\} \\ \delta(Side(T) \cup \Gamma(R)) & \text{iff } T \in \{N, E, S, W\} \\ U_{CDC} & \text{iff } T = B \end{cases} \quad (5.8)$$

Proof. The theorem is trivially proven combining the results of Theorem 5.4.1, Theorem 5.4.5, and Theorem 5.4.6. \square

Theorem 5.4.8. *Let O_1^* be an imprecise spatial-region object with non-empty yolk and O_2^+ be an imprecise spatial-region object having empty yolk, the crisp CDC disjunctive*

relation between them is computed as:

$$\begin{aligned} (i) \quad & R_{CDC}(O_1^*, O_2^+) \subseteq R_{CDC}(O_1^+, O_2^+) \\ (ii) \quad & \forall R_E \in R_{CDC}(O_1^*, O_2^+), \exists R_Y \in R_{CDC}(O_1^-, O_2^+) : \Gamma(R_E) \supseteq \Gamma(R_Y) \end{aligned} \quad (5.9)$$

Proof. This theorem directly follows from Theorem 5.4.2 and Theorem 5.4.7. \square

Theorem 5.4.9. Let O_1 be a precise object, O_2^* be an imprecise spatial-region object, and being $R_Y \triangleq R_{CDC}(O_1, O_2^-)$, $R_{E^\bullet} \triangleq R_{CDC}(O_1, O_2^\bullet)$, $T_{EY} \triangleq \Gamma(R_{E^\bullet}) \cap \Gamma(R_Y)$, and $R_{E^+} \triangleq R_{CDC}(O_1, O_2^+)$, the crisp CDC disjunctive relation between O_1 and O_2^* is:

$$R_{CDC}(O_1, O_2^*) = R_Y \quad \text{iff} \quad \Gamma(R_Y) = \Gamma(R_{E^\bullet}) \quad (5.10)$$

$$R_{CDC}(O_1, O_2^*) \subseteq \delta(\Gamma(R_Y) \cup \Gamma(R_{E^\bullet})) \quad \text{iff} \quad T_{EY} = \emptyset \quad (5.11)$$

$$R_{CDC}(O_1, O_2^*) \subseteq \{R \mid R \in R_{E^+} \wedge \Gamma(R) \supseteq T_{EY}\} \quad \text{iff} \quad T_{EY} \neq \emptyset \quad (5.12)$$

Proof. The theorem is trivially proven by combining the results of Equation 5.1, Theorem 5.4.5, and Theorem 5.4.6. \square

Theorem 5.4.10. Let O_1^+ be an imprecise spatial-region object with empty yolk, let O_2^* be an imprecise spatial-region object, and being $R_Y \triangleq R_{CDC}(O_1^\bullet, O_2^-)$, $R_{E^\bullet} \triangleq R_{CDC}(O_1^\bullet, O_2^\bullet)$, the crisp CDC disjunctive relation between them is:

$$R_{CDC}(O_1^+, O_2^*) = \delta(\Gamma(R_Y) \cup \Gamma(R_{E^\bullet})) \quad (5.13)$$

Proof. This theorem directly follows from Theorem 5.4.7 and Theorem 5.4.9. \square

Theorem 5.4.11. Being O_1^* and O_2^* two imprecise spatial-region objects, the crisp CDC relation between them is computed as:

$$\begin{aligned} (i) \quad & R_{CDC}(O_1^*, O_2^*) \subseteq R_{CDC}(O_1^+, O_2^*) \\ (ii) \quad & \forall R_E \in R_{CDC}(O_1^*, O_2^*), \exists R_Y \in R_{CDC}(O_1^-, O_2^*) : \Gamma(R_E) \supseteq \Gamma(R_Y) \end{aligned} \quad (5.14)$$

Proof. The theorem is trivially proven combining the results of Theorem 5.4.8 and Theorem 5.4.10. \square

5.4.3 Discussion

Two distinct approaches to model cardinal direction relations between imprecise objects have been proposed by Cicerone & Di Felice (2000) and Du & Guo (2010), that consider regions with a broad boundary (Clementini & Di Felice, 1996). Both the egg-yolk and the broad boundary are three-valued logics approaches to represent regions imprecisely

Table 5.4: Qualification of CDC relations between spatial-region objects.

Imprecise Relation	Example	Primary Object	Solution	Result
$R_{CDC}(O_i^+, O)$	Fig. 5.5(a)	O_1^+	Eq. 5.2	N
		O_2^+	Eq. 5.2	NE, E, NE:E
		O_3^+	Eq. 5.2	E, SE, S, E:SE, SE:S, E:SE:S
		O_4^+	Eq. 5.2	W, B, W:B
$R_{CDC}(O_i^*, O)$	Fig. 5.5(b)	O_1^*	Eq. 5.3	N
		O_2^*	Eq. 5.3	NE:E
		O_3^*	Eq. 5.3	SE:S, E:SE:S
		O_4^*	Eq. 5.3	W, W:B
$R_{CDC}(O_i, O^+)$	Fig. 5.5(c)	O_1	Eq. 5.5	N, NE, NW, N:NE, N:NW, N:NE:NW
		O_2	Eq. 5.4	NE
		O_3	Eq. 5.7	SE, E:SE, NE:E:SE
		O_4	Eq. 5.7	$U_{CDC} \setminus \delta(N, NE, E, SE, S, B)$
		O_5	Eq. 5.7	SE:S:SW
$R_{CDC}(O_i^+, O^+)$	Fig. 5.5(d)	O_1^+	Eq. 5.8	N, NE, NW, N:NE, N:NW, N:NE:NW
		O_2^+	Eq. 5.8	NE
		O_3^+	Eq. 5.8	NE, E, SE, NE:E, E:SE, NE:E:SE
		O_4^+	Eq. 5.8	U_{CDC}
		O_5^+	Eq. 5.8	SE, S, SW, SE:S, S:SW, SE:S:SW
$R_{CDC}(O_i^*, O^+)$	Fig. 5.5(e)	O_1^*	Eq. 5.9	N, NE, NW, N:NE, N:NW, N:NE:NW
		O_2^*	Eq. 5.9	NE
		O_3^*	Eq. 5.9	SE, E:SE, NE:E:SE
		O_4^*	Eq. 5.9	SW, S:SW, SE:S:SW
$R_{CDC}(O_i, O^*)$	Fig. 5.5(f)	O_1	Eq. 5.10	N
		O_2	Eq. 5.10	NE
		O_3	Eq. 5.12	E, E:SE
		O_4	Eq. 5.12	SW:W, S:SW:W
		O_5	Eq. 5.10	SE:S
		O_6	Eq. 5.12	W, NW, W:NW
$R_{CDC}(O_i^+, O^*)$	Fig. 5.5(g)	O_1^+	Eq. 5.13	N
		O_2^+	Eq. 5.13	NE
		O_3^+	Eq. 5.13	E, SE, E:SE
		O_4^+	Eq. 5.13	SE, S, SE:S
$R_{CDC}(O_i^*, O^*)$	Fig. 5.5(h)	O_1^*	Eq. 5.14	NE, E, NE:E
		O_2^*	Eq. 5.14	E, E:SE
		O_3^*	Eq. 5.14	SE, SE:S

described (Section 2.3.1); thus, it is possible to compare the results of Cicerone & Di Felice (2000) and Du & Guo (2010) with the method discussed above.

Cicerone & Di Felice (2000) propose an extension of the direction relation matrix (cf. Section 2.2.3.2). The plane is split into 25 tiles that result from the partitioning around the inner boundary (corresponding to the yolk) and the one around the outer boundary (corresponding to the egg). A matrix 5×5 is used to represent cardinal direction relations between regions with a broad boundary. The elements of the matrix can assume four different values, based on whether the inner region or the broad boundary fall into a tile. This approach is suited to represent 25^4 potential *fuzzy relations* between imprecise objects and does not yield any information about the crisp relations between the precise instances of the imprecise objects.

Differently, Du & Guo (2010) model cardinal direction relations between uncertain objects with broad boundary as the combination of four crisp direction relations, namely R_1 is the relation between the inner regions of the two objects, R_2 between the inner region of the reference object and the outer region of the primary one, R_3 is the relation holding between the outer region of the reference with the inner region of the primary, and finally R_4 is the relation holding between the outer regions. The cardinal direction relation between imprecise objects is then represented as $R = \langle R_1, R_2, R_3, R_4 \rangle$. For instance, the relation between O_1^* and O^* in Fig. 5.5(h) is $R = \langle \{NE\}, \{NE\}, \{E\}, \{NE:E\} \rangle$. In the case if one object is precise and the other is an object with broad a boundary, only two relations are sufficient to describe the cardinal direction relation. Furthermore, Du & Guo (2010) define the semantics of the four different relations in describing the relation. In particular, R_1 defines the *crisp part* of the relation, while R_2, R_3 and R_4 define its *uncertain part*. The crisp part of the relation represents those tiles in which the primary object has always to fall, while the uncertain part is the list of tiles in which it can fall. Therefore, the concept of *possibly north*, *possibly north-east*, etc. is defined to describe the uncertain part of the relation. Even though the approach proposed by Du & Guo (2010) is similar to the method described in this text, it does not always yield a correct result. For instance, considering again the objects O_1^* and O^* in Fig. 5.5(h), the crisp part of the relation is $\{NE\}$, while its uncertain part results to be $\{E\}$. This means that the relation is *north-east* and *possibly east*; this result is not correct since the relation can be also *east* as shown in Table 5.4, and in this case the primary object does not cross the *NE* tile. Moreover, the approach of Du & Guo (2010) does not allow for objects having empty inner region, and it is not hence suitable to model the relations between the objects in Fig. 5.5(a)-(c)-(d)-(e)-(g).

However, the proposed theory does not always yield disjunctive relations as precise as the propositions defined at the beginning of this section do (p. 115). In Fig. 5.6 two objects O_1^+ and O_2^+ are depicted both crossing $A_B^+(O^\circ)$ and $A_W^+(O^\circ)$. Hence, from Theorem 5.4.7 it results that $R_{CDC}(O_1^+, O^+) = U_{CDC}$ and also $R_{CDC}(O_2^+, O^+) = U_{CDC}$. In the former case it is possible, for any relation in U_{CDC} , to identify a precise instance of O_1^+ and a precise instance of O^+ that satisfy the relation. The same does not always hold true in the latter case: it is not possible, for instance, to find a precise instance O_2° of O_2^+ such that there exists O° precise instance of O^+ for which the relation $NE(O_2^\circ, O^\circ)$ holds. Instead, Proposition 5.4.2 yields a disjunctive crisp relation between O_2^+ and O^+ that does not admit NE . Anyway, the propositions discussed at the beginning of this section require the computation of a high number of quantifications, while the theorems only require the computation of a smaller set of them¹.

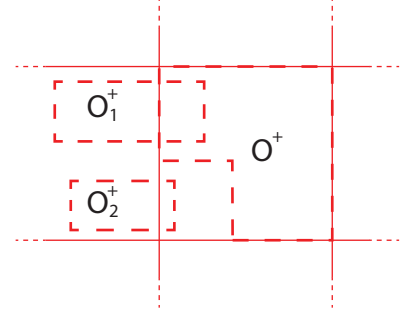


Figure 5.6: Objects configuration.

5.5 Computational Complexity of Qualification

The theoretical computational time complexity of the algorithms developed above for the qualification of crisp disjunctive relations between imprecise objects will be analyzed in this section.

5.5.1 Topology Qualification

The qualification of the crisp topological relation between imprecise objects is performed as shown in Section 5.2 by computing the topological base relations satisfied by the eggs and the yolks of the primary and reference spatial-region objects. Let n_1 and n_2 be the number of vertices that define respectively O_1^+ and O_1^- , let m_1 and m_2 be the number of vertices of O_2^+ , O_2^- , and let $N_{i,j} \triangleq n_i + m_j$ with $i, j \in \{1, 2\}$. The four topological relations between O_1^* and O_2^* are computed, in the worst case, in $\mathcal{O}(\sum_{i,j \in \{1,2\}} N_{i,j} \log N_{i,j})$ time (Schneider, 2002). The crisp topological relation is then retrieved by a lookup operation on either Table 5.2 or Table 5.1, and requires a constant time. Thus, the overall computational time complexity for the computation of a topological crisp relation between imprecise objects is $\mathcal{O}(\sum_{i,j \in \{1,2\}} N_{i,j} \log N_{i,j})$ in the worst case.

¹The empirical evaluation of the computation time as well as of the number of cases in which the theorems yield relations not as precise as the propositions do will be shown in Section 7.2.2.2.

5.5.2 Visibility Qualification

Proposition 5.3.1, Proposition 5.3.2 and Proposition 5.3.3 show how the visibility crisp relation between imprecise objects can be computed—respectively in the cases if the primary object is either precise or imprecise with empty yolk, or imprecise with non-empty yolk—by identifying which quantification of the visibility relations are crossed by the primary object.

Precise Primary Object

Let O_1 be a precise object defined by p vertices, let O_2^* be a spatial-region object whose egg is defined by n_1 vertices and whose yolk by n_2 vertices ($n = n_1 + n_2$), and let O_3^* be a spatial-region objects for which m_1, m_2 are the vertices that define respectively its egg and its yolk ($m = m_1 + m_2$). The visibility crisp relation between the three objects is computed by Proposition 5.3.1. The computation requires the following steps:

1. The computation, for any visibility base single-tile relation $R \in \mathcal{R}_{Vis}^{ST}$, of the quantifications $A_R^+(O_2^*, O_3^*)$. As shown in Section 4.7.2, this operation runs in $\mathcal{O}(n^2 + m)$ time in the worst case.
2. The verification, for any $R \in U_{Vis}$, of whether O_1 crosses $A_R^+(O_2^*, O_3^*)$. Since any A_R^+ is defined by $\mathcal{O}(n)$ vertices in the worst case, this operation requires $\mathcal{O}(pn)$ time in the worst case.

Overall, the proposed procedure to qualify $R_{Vis}(O_1, O_2^*, O_3^*)$ runs in $\mathcal{O}(n^2 + m + pn)$ time in the worst case.

Imprecise Primary Object with Empty Yolk

Let p be the number of vertices that define an imprecise object with empty yolk O_1^+ , let O_2^* be a spatial-region object whose egg is defined by n_1 vertices and whose yolk by n_2 vertices ($n = n_1 + n_2$), and let O_3^* be a spatial-region objects for which m_1, m_2 are the vertices that define respectively its egg and its yolk ($m = m_1 + m_2$). The visibility crisp relation is computed, in this case, through the equations shown in Proposition 5.3.2. The computation requires the same steps as in the precise case, thus $R_{Vis}(O_1^+, O_2^*, O_3^*)$ runs in $\mathcal{O}(n^2 + m + pn)$ time in the worst case.

Imprecise Primary Object

Let O_1^* be a spatial-region object whose egg is defined by p_1 vertices and whose yolk has p_2 vertices ($p = p_1 + p_2$), let O_2^* be a spatial-region object whose egg is defined by n_1 vertices and whose yolk by n_2 vertices ($n = n_1 + n_2$), and let O_3^* be a spatial-region object for which m_1, m_2 are the vertices that define respectively its egg and its yolk ($m = m_1 + m_2$). From Proposition 5.3.3, the qualification of the crisp visibility disjunctive relation between the three objects requires:

1. The identification of the relation $R_E = R_{Vis}(O_1^+, O_2^*, O_3^*)$. As shown above, this runs in $\mathcal{O}(n^2 + m + p_1n)$.
2. The computation of $R_Y = R_{Vis}(O_1^-, O_2^*, O_3^*)$. This step requires $\mathcal{O}(n^2 + m + p_2n)$ time in the worst case.
3. The combination of the achieved results. This operation requires a constant time.

Hence, the overall computational time complexity for computing $R_{Vis}(O_1^*, O_2^*, O_3^*)$ with the discussed algorithms is, in the worst case, in $\mathcal{O}(n^2 + m + pn)$.

5.5.3 Cardinal Direction Qualification

The worst case in the qualification of CDC crisp relation between imprecise objects is when both the objects are imprecise with non-empty yolk. Hence, two spatial-region objects O_1^*, O_2^* are considered: O_1^+ is defined by n_1 vertices, O_1^- by n_2 vertices ($n = n_1 + n_2$), and m_1, m_2 are the number of vertices that define respectively O_2^+ and O_2^- ($m = m_1 + m_2$).

From the first condition of Theorem 5.14, the relation $R_{CDC}(O_1^+, O_2^*)$ has to be computed; in turns, from Theorem 5.4.10, this requires the computation of the relations between precise objects $R_{CDC}(O_1^\bullet, O_2^-)$ and $R_{CDC}(O_1^\bullet, O_2^\bullet)$. The second condition of Theorem 5.14, instead, requires the qualification $R_{CDC}(O_1^-, O_2^*)$, that from Theorem 5.4.9 can be computed combining the relations between precise objects $R_{CDC}(O_1^-, O_2^-)$ and $R_{CDC}(O_1^-, O_2^\bullet)$. Summarizing, to compute $R_{CDC}(O_1^*, O_2^*)$, four relations between precise objects has to be computed (Eq. 5.1), and they are opportunely combined in order to derive the crisp relation between imprecise objects. Hence, the steps to perform are:

1. Computation of the quantifications $A_R^+(O_2^-)$. As shown in Section 4.7, this operation runs in $\mathcal{O}(n_2)$ time in the worst case.
2. Computation of the quantifications $A_R^+(O_2^\bullet)$. This operation runs in $\mathcal{O}(n_1)$ time in the worst case (cf. Section 4.7).
3. Intersection of O_1^- with the results of Step 1. Since the quantification of single-tile relation over a precise object has, in the worst case, four vertices, this step is performed in $\mathcal{O}(m_2)$ time in the worst case.
4. Intersection of O_1^- with the results of Step 2. As in Step 3, this operation requires $\mathcal{O}(m_2)$ time in the worst case.
5. Intersection of O_1^\bullet with the results of Step 1. This operation runs in $\mathcal{O}(m_1)$.
6. Intersection of O_1^\bullet with the results of Step 2. This operation requires $\mathcal{O}(m_1)$ time in the worst case.
7. Combination of the results obtained in Steps 3-4-5-6. This operation runs in constant time.

Overall, the qualification of a CDC crisp relation between imprecise objects requires $\mathcal{O}(n + m)$ time in the worst case.

5.6 Summary

In this chapter, the qualification of crisp relations between imprecise objects has been presented for topological, visibility and cardinal direction relations. Different approaches for the qualification have been adopted for the different calculi, based on their properties.

At first, the computation of the topological crisp relation has been grounded on the topological properties of the egg-yolk approach. Indeed, it is known that the yolk is always contained in the egg. Hence, to check whether a relation is admissible for two imprecise objects, a constraint network can be defined that describes the relations satisfied by the eggs and the yolks of the two objects. Reasoning algorithms are performed to check the consistency of the network and thus the admissibility of the relation. This approach is compatible with the approach of Cohn & Gotts (1996), moreover it allows for objects with empty yolk.

Visibility relations are qualified by checking which quantifications (cf. Chapter 4) are crossed by the primary object. This approach gives a generic rule to quantify relation based on the quantification outcomes, and it can be straightforwardly applied also to other calculi. Furthermore, at the best of the author's knowledge it is the first qualification approach for relations between imprecise objects for a ternary calculus.

Finally, even though the cardinal direction qualification can adopt the same approach defined for the visibility case, a theory has been defined to compute the crisp relation based only on the relations between the eggs and the yolks interpreted as precise objects. This allows for a reduction of the computational time complexity of the algorithm in the CDC case.

Chapter 6

A Hybrid Spatial Reasoning System

A system for the integration of spatial information grounded on *translation* operations has been analyzed in the previous chapters. This chapter enforces the integration process by adding reasoning capabilities to the system. An extended geographic information integration layer is presented in Section 6.1¹. Its components are detailed in Sections 6.2-6.6. Finally, a further extension to the system that takes *thematic* spatial information into consideration is introduced in Section 6.7.

6.1 A Hybrid Qualitative-Quantitative Reasoning System

In this section, a hybrid qualitative-quantitative reasoning system is introduced which exploits the individual strengths of computational geometry based inference, in particular polygon intersection and union, and relation algebraic qualitative reasoning methods. The goal is to be able to perform spatial inference in mixed settings. This means, input entities are partially described geometrically in the form of imprecise objects, while others are described using qualitative relations from different qualitative spatial calculi, either with respect to the geometrically defined objects or with respect to other qualitatively described objects. The central reasoning task is to determine approximations for the geometries of input objects for which no exact geometry is given as well as to derive more precise qualitative information.

¹Part of the work described in this chapter has been presented in De Felice *et al.* (2011).

6.1.1 Architecture

The geographic information integration layer introduced in Section 3.2.3 —whose preliminary design has been shown in Section 4.1—is extended here by the inclusion of reasoning capabilities into the system. The hybrid reasoning approach that combines qualitative and geometric reasoning components with qualification and quantification components is shown in Fig. 6.1.

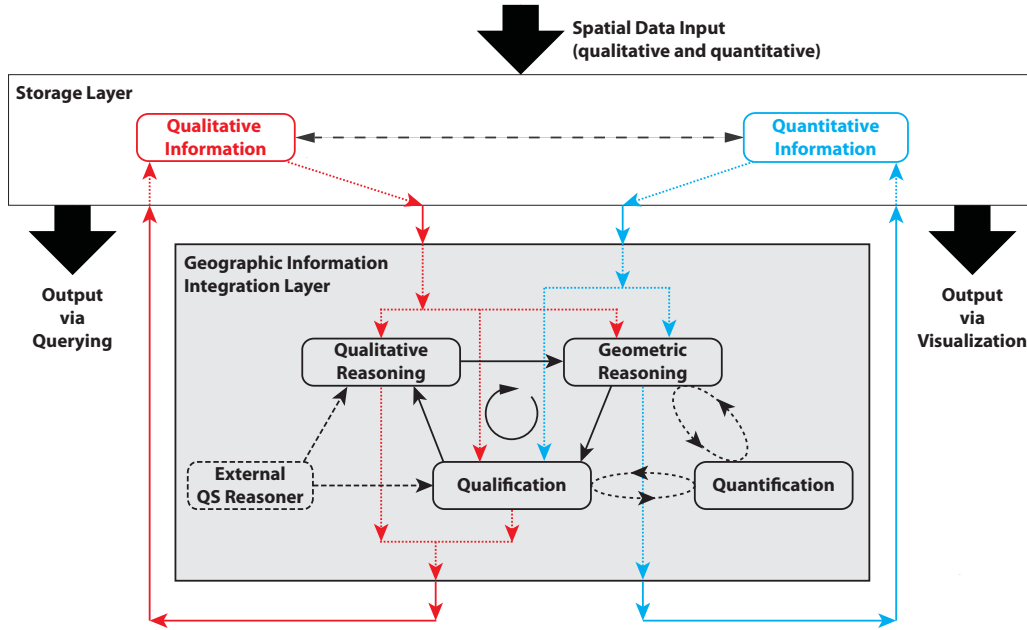


Figure 6.1: Architecture of the hybrid qualitative-quantitative reasoning system¹.

The quantification and qualification component perform the mediation between the quantitative and qualitative representations used in the two reasoning components in both directions: The quantification procedure computes the quantitative interpretation of a qualitative relation taking into account the geometries of the reference spatial-region objects as far as they are known (cf. Chapter 4). It is called from within the geometric reasoning and qualification components. The qualification procedure derives qualitative spatial relations from the geometries of the involved spatial-region objects (cf. Chapter 5). It is called to translate the output of the geometric reasoning and its

¹The red lines in the figure represent the flow of qualitative information: solid lines show the stream of information among the different layers of the system, while dotted ones depict how the information flows within any single layer. In the same way, the cyan lines represent the stream of quantitative information. The dashed black lines within the integration show functional dependencies among the system's components. Finally, solid black lines are used to represent the reasoning system's cycle. The roles of the different layers have been described in Fig. 3.5.

output then forms the input for the qualitative reasoner. By alternately executing the geometric and qualitative reasoning components, the overall reasoner is able to deduce new spatial information about the input objects both on the qualitative side (new or more specific qualitative relations) as well as on the quantitative side (more precise minimal and maximal approximations). The overall reasoning cycle terminates when a fixpoint has been reached where neither more specific geometries nor more specific qualitative relations can be deduced. The qualitative reasoning and qualification modules can also resort to functionalities provided by existing QSR software, such as SparQ (Wallgrün *et al.*, 2007) and GQR (Gantner *et al.*, 2008) to perform fast constraint propagation.

In the remainder of this chapter, the combined qualitative-quantitative reasoning cycle is shown in detail using RCC-8, CDC, and visibility (cf. Section 2.2.3) as exemplary calculi. Section 6.2 shows how qualitative information is represented in a structured way by extending the concept of constraint network (cf. Section 2.2.2). The geometric reasoning component is presented in Section 6.3. It approximates the extension of every unknown entity by combining the quantifications of all relations it is involved in. An extension of the qualification procedure (see Chapter 5) is discussed in Section 6.4. It qualifies the relations holding between any permutation of the objects in the system. Section 6.5 shows the qualitative spatial reasoning component. Finally, the hybrid reasoning algorithm is described in Section 6.6.

6.2 Multi-calculus Constraint Network

A configuration of objects can be qualitatively represented by a constraint network, as shown in Section 2.2.2. However, a constraint network is defined for a single calculus C , while in this work more than one calculus is considered at the same time. Hence, a *multi-calculus constraint network* is defined. It is a constraint network whose edges are labeled by tuples of relations from different calculi.

Definition 20 (Multi-calculus constraint network MN).

Let $\mathcal{C} = \{C_1, \dots, C_a\}$ be a set of qualitative spatial calculi, and $\mathcal{R}_{\mathcal{B}_{C_i}}$ be the set of general relations defined in the calculus $C_i \in \mathcal{C}$. A multi-calculus constraint network MN is defined as a triple (V, \mathcal{D}, MR) with variables $V = \{O_1^*, \dots, O_n^*\}$ over a domain \mathcal{D} , whose valuation is constrained by the multi-dimensional constraint matrix MR . Every element of MR —denoted as $MR_{O_i^*, \dots, O_{i+m}^*} = \langle R_{C_1}, \dots, R_{C_a} \rangle$, with $R_{C_i} \in \mathcal{R}_{\mathcal{B}_{C_i}}$ —describes the m -ary constraint between O_i^*, \dots, O_{i+m}^* .

As an example, the configuration of objects drawn in Fig 6.2(a) is represented by the multi-calculus constraint network described in Fig. 6.2(b), which considers the

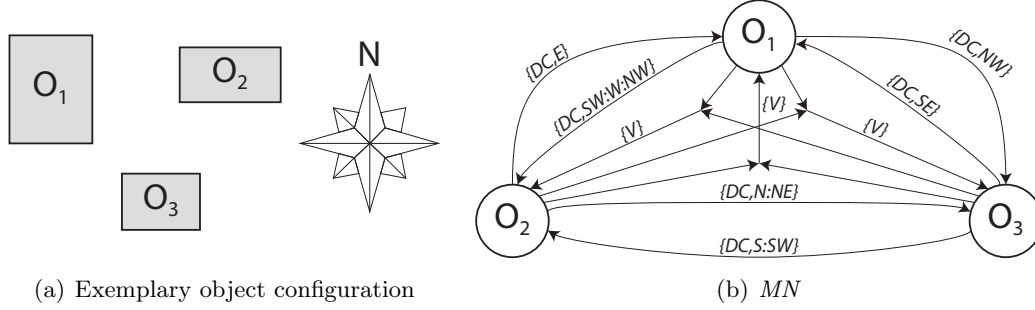


Figure 6.2: Multi-calculus constraint network.

set of calculi \mathcal{C} composed by RCC-8, CDC, and visibility calculi. The multi-calculus constraint network is represented as a hyper-graph, since its edges can connect more than two nodes. Indeed, the MN in Fig. 6.2(b) has some edges with one input node and one output node—that represent the binary constraints given by topological and directional relations—and hyper-edges having two input nodes and one output node—that describe the ternary visibility constraints. For the sake of simplicity, the MN in the example only contains one ternary hyper-edge for every pair of input nodes. In fact, the real network has two ternary hyper-edges for every pair of reference objects O_i^* and O_j^* : one for the constraint in which the reference objects are the ordered pair $\langle O_i^*, O_j^* \rangle$, and one for the constraint having $\langle O_j^*, O_i^* \rangle$ as reference.

Let MN be a multi-calculus constraint network that describes a configuration of objects through a set of calculi \mathcal{C} . MN_C denotes the constraint network that describes the configuration with respect to the calculus C only. For instance, Fig 6.3 shows the constraint networks for the calculi RCC, CDC, and visibility extracted from the multi-calculus network in Fig. 6.2(b).

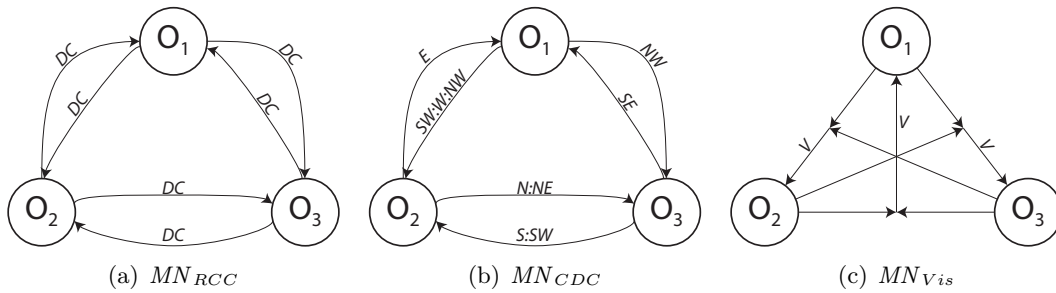


Figure 6.3: MN_C derived from the multi-calculus constraint network in Fig. 6.2.

The function $refineRelation(MN_C, (O_1^*, \dots, O_n^*), R)$ is defined that refines the relation between the objects O_1^*, \dots, O_n^* in the network MN_C by intersecting the previous relation for calculus C with the relation R . It yields a boolean value that is set to *True*

if the constraint is refined. Furthermore, the function $QualitativeRelations(O_i^*, MN)$ is defined that gets as input a spatial-region object O_i^* and a multi-calculus constraint network MN and returns all relations in MN whose primary object is O_i^* . The relations are returned as a list of tuples defined as $\langle R, C, O_2^*, \dots, O_n^* \rangle$, where R is the relation in the calculus C satisfied between O_i^* and the reference objects O_2^*, \dots, O_n^* .

6.3 The Geometric Reasoning Component

The geometric reasoning component computes *quantitative approximations* for all imprecise spatial-region objects that correspond to the unknown entities of the system. It uses intersection and union procedures for drawing geometric inferences by combining information stemming from different relations. This results in refined quantitative approximations.

Algorithm 7 *GeometricReasoner*(G, MN)

```

hasChanged  $\leftarrow$  True
while hasChanged do
  hasChanged  $\leftarrow$  False
  for  $O_i^* \in G$  do
     $PR_i^* \leftarrow O_i^*$ 
     $QR \leftarrow QualitativeRelations(O_i^*, MN)$ 
    for  $\langle R, C, O_2^*, \dots, O_n^* \rangle \in QR$  do
       $O^* \leftarrow Quantify(R, C, O_2^*, \dots, O_n^*)$ 
       $O_i^+ \leftarrow O_i^+ \cap O^+$ ;  $O_i^- \leftarrow O_i^- \cup O^-$ 
    end for
    if  $PR_i^* \neq O_i^*$  then hasChanged  $\leftarrow$  True end if
  end for
end while
return  $G$ 

```

The pseudocode for the geometric reasoning procedure is given in Algorithm 7. It takes the set G containing the current quantitative approximations for all involved objects in egg-yolk format and the multi-calculus constraint network MN . It continuously loops through all spatial-region objects $O_i^* \in G$, considers all qualitative relations this object is involved in as provided by the function $QualitativeRelations$, and tries to improve the quantitative description O_i^* based on these relations. To achieve this, the $Quantify(R, C, O_2^*, \dots, O_n^*)$ procedure previously defined in Algorithm 3 (see page 76) is called. The returned imprecise object O^* is used to update the object $O_i^* \in G$. For the maximal extension O_i^+ this has to be done by taking the intersection of the previous value of O_i^+ and O^+ . Conversely, for the minimal extension O_i^- the combination has to be done with the union operator.

When the geometric approximation of an object O_i^* is refined, this can mean that now approximations can be refined further for objects O_j^* that stand in certain relations with O_i^* . Therefore, the algorithm runs the loop through all objects until no refined quantitative description can be computed. To keep track of this, the boolean flag *hasChanged* is used and the previous quantitative description O_i^* is stored in PR_i^* and later compared to the newly computed approximation. The object set G containing the refined geometries is returned by the reasoning algorithm.

6.3.1 Computational Complexity of Geometric Reasoning

Let n be the number of elements in the set G , the constraint network MN has n nodes and $\mathcal{O}(n^m)$ arcs, where m corresponds to the maximal arity of the calculi represented in the multi-calculus constraint network. It is assumed that q is the time required to execute the function *Quantify*¹.

For any spatial-region object O_i^* in G , there exist $\mathcal{O}(n^{m-1})$ relations in MN for which O_i^* acts as primary object. Thus, the quantification of a single object costs $\mathcal{O}(n^{m-1}q)$ time in the worst case. The time required for the quantification of all objects in G is $\mathcal{O}(n^m q)$. Quantification of all spatial-region objects is repeated until the geometric descriptions of the objects in G do not change anymore. The worst case occurs when only one object is refined during any iteration, hence the external cycle is repeated at most n times. Overall, the *GeometricReasoner* algorithm (Algorithm 7) runs in $\mathcal{O}(n^{m+1}q)$ time in the worst case. For the calculi considered in this work (binary and ternary) the computational complexity becomes $\mathcal{O}(n^4 q)$ time in the worst case.

6.4 The Qualification Component

Qualification is the procedure that determines the qualitative spatial crisp relations holding between the spatial-region objects $O_i^* \in G$ for a set of calculi \mathcal{C} . The pseudocode of the qualifier procedure is given in Algorithm 8. Besides G , the function takes as input the multi-calculus constraint network MN which it modifies by refining the constraints based on the newly computed relations. The resulting network is then returned as the result of the qualification. The algorithm computes the relations holding between any permutation of spatial-region objects in G ; all m -ary permutations are calculated by an auxiliary function *Permutations*(G, m). The function *refineRelation* is called with the relation returned by the function *Qualify_C* described in Chapter 5.

¹As shown in Section 4.7, the computational time complexity of *Quantify* strictly depends on the calculi that are considered.

Algorithm 8 *Qualify*(G, MN)

```

for  $C \in \mathcal{C}$  do
   $Perm \leftarrow \text{Permutations}(G, \text{arity}(C))$ 
  for  $(O_1^*, \dots, O_n^*) \in Perm$  do
     $\text{refineRelation}(MN_C, (O_1^*, \dots, O_n^*), \text{Qualify}_C(O_1^*, \dots, O_n^*))$ 
  end for
end for
return  $MN$ 

```

6.4.1 Computational Complexity of the Qualification Component

Let n be the number of elements in G . The constraint network MN has n nodes and $\mathcal{O}(n^m)$ arcs, where m corresponds to the maximal arity of the relations represented in the multi-calculus constraint network. It is further assumed that q is the time required to run the function Qualify_C . The number of m -permutations of the set G is $\binom{n}{m} < n^m$. Furthermore, the *refineRelation* function runs in constant time¹. Hence, it directly follows that Algorithm 8 runs in $\mathcal{O}(|\mathcal{C}|qn^m)$ time in the worst case. For the calculi considered in this work, the computational complexity becomes $\mathcal{O}(|\mathcal{C}|qn^3)$ time in the worst case.

6.5 The Qualitative Reasoning Component

The qualitative reasoning component of the hybrid reasoning system is founded on the *algebraic closure* algorithm (Mackworth, 1977; Montanari, 1974) that applies composition and permutation operations defined in a calculus until a fixpoint has been reached. The function *AlgebraicClosure*(C, G, MN_C) in the *QualitativeReasoner* algorithm (Algorithm 9) performs algebraic closure for a calculus C on a constraint network MN_C . The *AlgebraicClosure* procedure picks the right relations from the tuples for the given calculus C and modifies them. The main algorithm loops through all calculi C in a given set of calculi \mathcal{C} . It returns the potentially modified network MN . Let $G = \{O_1^*, \dots, O_n^*\}$, the indices $i, j, k, l \leq n$ will be used in the remainder of this section to identify elements in G .

Algorithm 9 *QualitativeReasoner*(G, MN)

```

for  $C \in \mathcal{C}$  do
   $MN_C \leftarrow \text{AlgebraicClosure}(C, G, MN_C)$ 
end for
return  $MN$ 

```

¹The multi-dimensional constraint matrix MR of the multi-calculus constraint network can be accessed by index in constant time.

The algebraic closure procedure—first introduced by Montanari (1974) and then improved by Mackworth (1977)—for a binary qualitative calculus is shown in Algorithm 10. A queue s is initialized with all pairs of objects involved in the qualitative relations defined in the constraint network N . Any constraint in the queue is propagated through permutation and (weak) composition operations. At first, the inverse operation— $R_{O_i^* O_j^*} \Rightarrow R_{O_j^* O_i^*}$ —is checked; if the result refines the edge in the network corresponding to $R_{O_j^* O_i^*}$, the edge is updated with the refined relation and (O_j^*, O_i^*) is added to the queue of pairs to analyze. Afterwards, for every $O_k^* \in G$, the compositions $R_{O_i^* O_j^*} \diamond R_{O_j^* O_k^*}$ and $R_{O_k^* O_i^*} \diamond R_{O_i^* O_j^*}$ are computed and their results are eventually propagated. The propagation process is repeated until it does not introduce any modification in the constraint network. Being n the number of nodes in the network, the *AlgebraicClosure* procedure performs in $\mathcal{O}(n^3)$ time in the worst case.

Algorithm 10 *AlgebraicClosure*(C, G, N) – Binary Constraints

Procedure *AlgebraicClosure*(C, G, N)

$s \leftarrow \{(O_i^*, O_j^*) \mid R_{O_i^* O_j^*} \in N\}$

while $s \neq \emptyset$ **do**

$s' \leftarrow \{\}$

for $(O_i^*, O_j^*) \in s$ **do**

$s', N \leftarrow \text{CheckPermutation}((O_i^*, O_j^*), s', N)$

for $O_k^* \in G$ **do** $s', N \leftarrow \text{CheckComposition}((O_i^*, O_j^*), O_k^*, s', N)$ **end for**

end for

$s \leftarrow s'$

end while

return N

Procedure *CheckPermutation*($(O_i^*, O_j^*), s', N$)

$hasChanged \leftarrow \text{refineRelation}(N, (O_j^*, O_i^*), R_{O_i^* O_j^*})$

if $hasChanged$ **then** $s' \leftarrow s' \cup (O_j^*, O_i^*)$ **end if**

return s', N

Procedure *CheckComposition*($(O_i^*, O_j^*), O_k^*, s', N$)

if $O_k^* \neq O_i^* \wedge O_k^* \neq O_j^*$ **then**

$hasChanged \leftarrow \text{refineRelation}(N, (O_i^*, O_k^*), R_{O_i^* O_j^*} \diamond R_{O_j^* O_k^*})$

if $hasChanged$ **then** $s' \leftarrow s' \cup (O_i^*, O_k^*)$ **end if**

$hasChanged \leftarrow \text{refineRelation}(N, (O_k^*, O_j^*), R_{O_k^* O_i^*} \diamond R_{O_i^* O_j^*})$

if $hasChanged$ **then** $s' \leftarrow s' \cup (O_k^*, O_j^*)$ **end if**

end if

return s', N

An example of the algebraic closure procedure is given in Fig. 6.4, where RCC-8 relations are used as constraints into the constraint network N^1 . The queue s

¹The inverse and composition operations of RCC-8 relations are depicted in the reasoning tables shown in Chapter 2 (respectively Table 2.1 p. 30 and Table 2.2 p. 30).

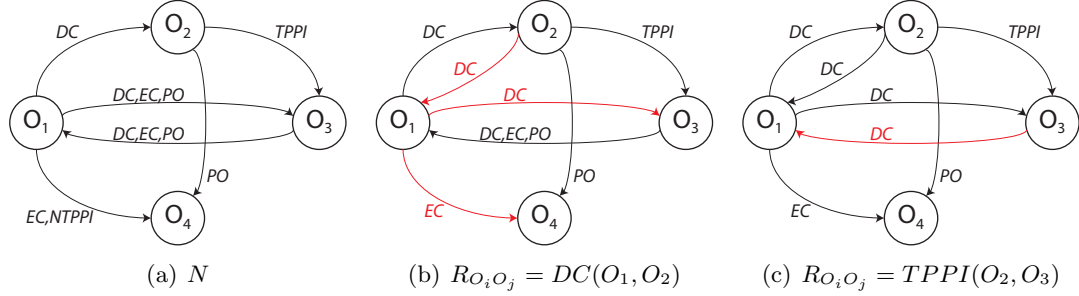


Figure 6.4: Algebraic closure algorithm.

gets all pairs of objects for which there exists a constraint in the network ($s \leftarrow \{(O_1, O_2), (O_2, O_3), \dots\}$) and any constraint is propagated at a time. At first, the relation $DC(O_1, O_2)$ is analyzed: The permutation operation yields $DC(O_2, O_1)$; the previous constraint between O_2 and O_1 is equal to \mathcal{B}_{RCC}^1 , hence the edge is refined by the algorithm and added to s' . Afterwards, the algorithm performs a cycle over all nodes in the network, and the respective constraints are composed. For instance, when $O_k = O_3$, the constraint $DC(O_1, O_2)$ is composed with $TPPI(O_2, O_3)$; the resulting relation is used to refine the arc between O_1 and O_3 , that becomes $DC(O_1, O_3)$ and the constraint is added to s' . Also $\{DC, EC, PO\}(O_3, O_1)$ is composed with $DC(O_1, O_2)$; however, the resulting relation— \mathcal{B}_{RCC} —does not refine the arc between O_3 and O_2 . Next, the object $O_k = O_4$ is considered. The composition of the constraints $DC(O_1, O_2)$ and $PO(O_2, O_4)$ yields a refinement of the edge between O_1 and O_4 (that is added to s'), while the other composition operation does not produce any refinement for the constraint network. The reduced network resulting after the propagation of the constraint $R_{O_1 O_2}$ is shown in Fig. 6.4(b). The next constraint in the network ($R_{O_1 O_3}$) is then taken into consideration; it yields the network drawn in Fig. 6.4(c) through the same process. Once all constraints have been propagated, s is replaced with s' and the propagation process starts again. This is repeated until no constraint in the network changes anymore.

The algebraic closure procedure has been adapted to deal with ternary calculi by Dylla & Moratz (2004); the pseudocode is shown in Algorithm 11. The algorithm performs all possible permutation operations of ternary relations². The *CheckPermu-*

¹As shown in Section 2.2.2, a missing arc in the constraint network stands for the universal relation U_C .

²The permutation operations of a ternary relation have been defined in Section 2.2.1.2. To summarize, they are: $ID(R_{O_i^* O_j^* O_k^*}) \rightarrow R_{O_i^* O_j^* O_k^*}$, $INV(R_{O_i^* O_j^* O_k^*}) \rightarrow R_{O_j^* O_i^* O_k^*}$, $SC(R_{O_i^* O_j^* O_k^*}) \rightarrow R_{O_i^* O_k^* O_j^*}$, $SCI(R_{O_i^* O_j^* O_k^*}) \rightarrow R_{O_k^* O_i^* O_j^*}$, $HM(R_{O_i^* O_j^* O_k^*}) \rightarrow R_{O_i^* O_k^* O_j^*}$, and $HMI(R_{O_i^* O_j^* O_k^*}) \rightarrow R_{O_j^* O_i^* O_k^*}$.

tation procedure in the algorithm uses the operator $R_{O_i^* O_j^* O_k^*}^p$ to indicate the result given by the permutation operation p . For instance, $R_{O_i^* O_j^* O_k^*}^{INV}$ yields $R_{O_j^* O_i^* O_k^*}$; instead, $INV(O_i^* O_j^* O_k^*)$ returns the list (O_j^*, O_i^*, O_k^*) . The composition operation used in the described algorithm is defined as $R_{O_1 O_2 O_4} = R_{O_1 O_2 O_3} \diamond R_{O_2 O_3 O_4}$. If n is the number of nodes in the network, the ternary algebraic closure procedure runs in $\mathcal{O}(n^4)$ time in the worst case.

However, the described procedure is not suitable for calculi which define a different composition operation and for weak calculi in which several composition operations need to be defined (i.e., the visibility calculus¹).

Algorithm 11 *AlgebraicClosure*(C, G, N) – Ternary Constraints

Procedure *AlgebraicClosure*(C, G, N)

$s \leftarrow \{(O_i^*, O_j^*, O_k^*) \mid R_{O_i^* O_j^* O_k^*} \in N\}$

while $s \neq \emptyset$ **do**

$s' \leftarrow \{\}$

for $(O_i^*, O_j^*, O_k^*) \in s$ **do**

$s', N \leftarrow \text{CheckPermutation}((O_i^*, O_j^*, O_k^*), s', N)$

for $O_l^* \in G$ **do** $s', N \leftarrow \text{CheckComposition}((O_i^*, O_j^*, O_k^*), O_l^*, s', N)$ **end for**

end for

$s \leftarrow s'$

end while

return N

Procedure *CheckPermutation*((O_i^*, O_j^*, O_k^*), s', N)

for $p \in \{ID, INV, SC, SCI, HM, HMI\}$ **do**

$hasChanged \leftarrow \text{refineRelation}(N, p(O_i^*, O_j^*, O_k^*), R_{O_i^* O_j^* O_k^*}^p)$

if $hasChanged$ **then** $s' \leftarrow s' \cup p(O_i^*, O_j^*, O_k^*)$ **end if**

end for

return s', N

Procedure *CheckComposition*((O_i^*, O_j^*, O_k^*), O_l^*, s', N)

if $O_l^* \neq O_i^* \wedge O_l^* \neq O_j^* \wedge O_l^* \neq O_k^*$ **then**

$hasChanged \leftarrow \text{refineRelation}(N, (O_i^*, O_j^*, O_l^*), R_{O_i^* O_j^* O_k^*} \diamond R_{O_j^* O_k^* O_l^*})$

if $hasChanged$ **then** $s' \leftarrow s' \cup (O_i^*, O_j^*, O_l^*)$ **end if**

$hasChanged \leftarrow \text{refineRelation}(N, (O_l^*, O_i^*, O_k^*), R_{O_l^* O_i^* O_j^*} \diamond R_{O_i^* O_j^* O_k^*})$

if $hasChanged$ **then** $s' \leftarrow s' \cup (O_l^*, O_i^*, O_k^*)$ **end if**

end if

return s', N

¹Tarquini *et al.* (2007) define the composition of visibility relations as $R(O_1, O_3, O_4) = R_1(O_1, O_2, O_3) \diamond R_2(O_2, O_3, O_4)$, while a different composition operation— $R(O_2, O_3, O_4) = R_1(O_1, O_2, O_3) \diamond R_2(O_1, O_3, O_4)$ —is presented by De Felice *et al.* (2010).

6.5.1 Generalization of Ternary Relations' Composition

In order to allow the algebraic closure procedure to deal with ternary calculi that define several composition operations, a generalization of composition of ternary relations is introduced and the *CheckComposition* procedure is accordingly modified.

Let C be a ternary calculus. A *composition rule* is defined as an ordered list of three elements, each of which is in turn a list of three elements in \mathbb{N} . The i -th composition rule in the calculus C is:

$$CRule_{C,i} = [CR_{i1} \mid CR_{i2} \mid CR_{i3}] = [\langle cr_{i11}, cr_{i12}, cr_{i13} \rangle \mid \langle cr_{i21}, cr_{i22}, cr_{i23} \rangle \mid \langle cr_{i31}, cr_{i32}, cr_{i33} \rangle]$$

In order to be *valid*, the composition rule has to satisfy the following constraints:

- $|\{cr_{i11}, cr_{i12}, cr_{i13}\} \cup \{cr_{i21}, cr_{i22}, cr_{i23}\}| = 4$
- $\{cr_{i31}, cr_{i32}, cr_{i33}\} \subset \{cr_{i11}, cr_{i12}, cr_{i13}\} \cup \{cr_{i21}, cr_{i22}, cr_{i23}\}$
- $\{cr_{i11}, cr_{i12}, cr_{i13}\} \cap \{cr_{i31}, cr_{i32}, cr_{i33}\} \neq \emptyset$
- $\{cr_{i21}, cr_{i22}, cr_{i23}\} \cap \{cr_{i31}, cr_{i32}, cr_{i33}\} \neq \emptyset$.

The first element of a composition rule parametrically denotes the list of objects that are in the first input relation, the second represents the objects in the second input relation, while the third element parametrically gives the ordering of the objects in the output relation. For instance, the composition rules for the visibility calculus are defined as:

$$CRule_{Vis,1} = [\langle 1, 2, 3 \rangle \mid \langle 2, 3, 4 \rangle \mid \langle 1, 3, 4 \rangle]$$

$$CRule_{Vis,2} = [\langle 1, 2, 3 \rangle \mid \langle 1, 3, 4 \rangle \mid \langle 2, 3, 4 \rangle]$$

...

Let $n, m \in \{1, 2\}, n \neq m$. A function $ComposeRel(CRule, CR_n, (O_i^*, O_j^*, O_k^*), O_l^*, N)$ is defined that computes the composition, as defined in the composition rule $CRule$, by assigning (O_i^*, O_j^*, O_k^*) to the elements of CR_n and O_l^* to the value of CR_m that is not in CR_n , and finally performing the composition $R_{CR_1} \diamond R_{CR_2}$. For instance, $ComposeRel(CRule_{Vis,2}, CR_1, (O_1, O_2, O_3), O_4, N)$ yields $R_{O_1, O_2, O_3} \diamond R_{O_1, O_3, O_4}$. Similarly, a function $ComposeObjects(CRule, CR_n, (O_i^*, O_j^*, O_k^*), O_l^*)$ is defined, that yields the tuple of objects resulting from the composition, by performing the same assignments as $ComposeRel$. As an example, $ComposeObjects(CRule_{Vis,2}, CR_1, (O_1, O_2, O_3), O_4)$ yields (O_2, O_3, O_4) .

The procedure *CheckComposition* is modified in order to deal with a set of composition rules, as shown in Algorithm 12. The modification does not change the computational time complexity of *AlgebraicClosure*, because the number of possible composition rules for a calculus C is limited.

Algorithm 12 *CheckComposition* $((O_i^*, O_j^*, O_k^*), O_l^*, s', N)$ – Ternary constraints

```

if  $O_l^* \neq O_i^* \wedge O_l^* \neq O_j^* \wedge O_l^* \neq O_k^*$  then
  for  $CRule_{C,i} \in CRule_C$  do
     $R \leftarrow ComposeRel(CRule_{C,i}, CR_1, (O_i^*, O_j^*, O_k^*), O_l^*, N)$ 
     $(O_a^*, O_b^*, O_c^*) \leftarrow ComposeObjects(CRule_{C,i}, CR_1, (O_i^*, O_j^*, O_k^*), O_l^*)$ 
     $hasChanged \leftarrow refineRelation(N, (O_a^*, O_b^*, O_c^*), R)$ 
    if  $hasChanged$  then  $s' \leftarrow s' \cup (O_a^*, O_b^*, O_c^*)$  end if
     $R \leftarrow ComposeRel(CRule_{C,i}, CR_2, (O_i^*, O_j^*, O_k^*), O_l^*, N)$ 
     $(O_a^*, O_b^*, O_c^*) \leftarrow ComposeObjects(CRule_{C,i}, CR_2, (O_i^*, O_j^*, O_k^*), O_l^*)$ 
     $hasChanged \leftarrow refineRelation(N, (O_a^*, O_b^*, O_c^*), R)$ 
    if  $hasChanged$  then  $s' \leftarrow s' \cup (O_a^*, O_b^*, O_c^*)$  end if
  end for
end if
return  $s', N$ 

```

6.6 The Hybrid Spatial Reasoning Algorithm

A basic pseudocode version of the procedure that combines the components described in the previous sections is given in Algorithm 13. To keep it simple, it does not contain all possible optimizations for avoiding unnecessary computation¹. The reasoner takes the set G of spatial-region objects for all involved spatial entities and a multi-calculus constraint network MN representing the input information. It is assumed that the geometries in G have been initialized in accordance with the input information and by using imprecise objects representing the plane² for entities for which no geometric information is given. The multi-calculus constraint network MN is also assumed to have been initialized correctly using the specified qualitative relations where possible, while using the universal relation (disjunction of all base relations) everywhere else.

At first, the qualitative reasoning component is applied to the input network MN , refining as many qualitative relations as possible. The resulting network is then passed on to the geometric reasoner that produces new geometric approximations (under the support of the quantification component). Afterwards, the approximated regions are analyzed by the qualification procedure to further refine the qualitative constraint network MN . This process is repeated until neither the qualitative nor the geometric reasoning component are able to refine the available information further.

Given that the input information is consistent, it follows that the hybrid reasoning is correct in the sense that computed geometric approximations have to indeed contain the actual object and that the actual geometric configuration described by the input is a solution of the derived qualitative model: (1) The *QualitativeReasoner* and *Qualify* functions can only lead to a refinement of the constraint network MN ; and (2) the

¹Some heuristics to avoid unnecessary computation will be discussed in Section 6.6.2.

²A way to represent the plane by means of infinite-region is detailed in Appendix A.

Algorithm 13 *HybridReasoner*(G, MN)

```

hasChanged  $\leftarrow$  True
while hasChanged do
  previousMN  $\leftarrow$  MN
  MN  $\leftarrow$  QualitativeReasoner( $G, MN$ )
   $G \leftarrow$  GeometricReasoner( $G, MN$ )
  MN  $\leftarrow$  Qualify( $G, MN$ )
  if  $MN = \textit{previousMN}$  then hasChanged  $\leftarrow$  False end if
end while
return ( $G, MN$ )

```

GeometricReasoner can only further refine the geometries derived from the qualitative relations. The overall procedure ends when either the *GeometricReasoner* cannot refine any geometry and, hence, *Qualify* cannot discover new qualitative relations, or when the *QualitativeReasoner* does not refine the constraint network, resulting in an absence of information that would enable the *GeometricReasoner* to further refine geometries. Furthermore, the maximal quantification of a relation is based on the regions in which an entity has to be contained, hence the maximal quantification represents an upper approximation for the geometry of the entity. Similarly, the minimal quantification represents a lower approximation for the geometry of the entity.

Considering only binary and ternary calculi, let n be the number of objects in G , Algorithm 13 performs in $\mathcal{O}(c(n^4 + |\mathcal{C}|qn^3 + n^4q)) = \mathcal{O}(cq(|\mathcal{C}|n^3 + n^4))$ time in the worst case, with c being the number of iterations performed by the hybrid reasoner procedure, and q being the time required to run the *Qualify_C* function. Since $|\mathcal{C}|$ is constant, the computational time complexity is in $\mathcal{O}(cqn^4)$ in the worst case.

6.6.1 A Hybrid Spatial Reasoning Example

An object configuration that is used in the remainder of this section for a step-by-step running example of the hybrid reasoning procedure described above is displayed in Fig. 6.5(a). In this example, it is supposed that the reasoner is given precise geometric definitions for regions O_1, O_2, O_3, O_7, O_8 (see Fig. 6.5(b)); moreover, the reasoner gets as input the following qualitative descriptions:

$$\begin{array}{lll}
 PV_J(O_6^*, O_4^*, O_2) & PV_J(O_5^*, O_4^*, O_2) & PV_L(O_5^*, O_2, O_3) \quad TPP(O_7, O_4^*) \\
 W(O_6^*, O_8) & N(O_6^*, O_1) & S(O_9^*, O_4^*)
 \end{array}$$

Note that some relations refer to spatial entities which geometric description is not known, for instance in $PV_J(O_6^*, O_4^*, O_2)$.

In the first step, the qualitative reasoner infers 9 new base relations and 6 refinements of disjunctive relations, e.g., $TPPI(O_4^*, O_7)$ and $\{PV_L, Oc\}(O_4^*, O_2, O_3)$. Feeding

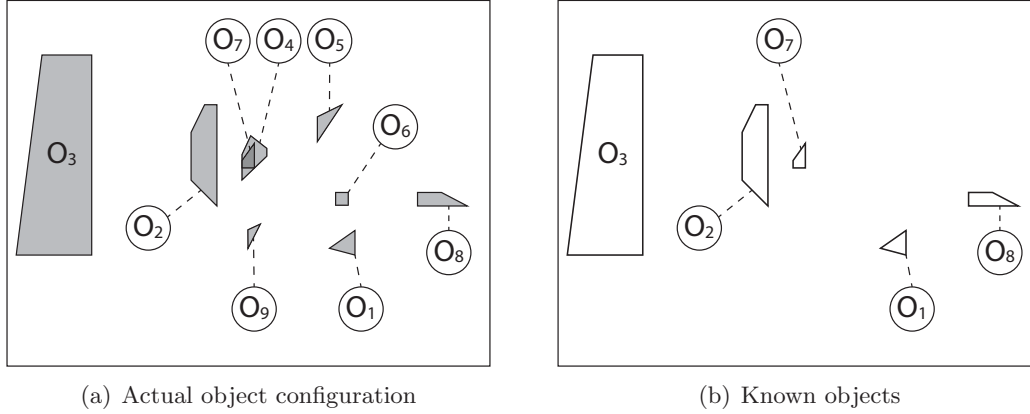


Figure 6.5: Actual object configuration and known objects.

the resulting qualitative relations into the geometric reasoner yields a first approximation of the objects that describe the unknown entities O_4^* , O_5^* , O_6^* and O_9^* , as depicted in Fig. 6.6. Note that object O_4^* is the only one having a yolk component O_4^- defined by the inferred relation $TPPI(O_4^*, O_7)$. The qualification procedure is now able to produce 253 new qualitative relations using both precise and imprecise spatial-region objects. The current system outcomes, in terms of quantitative descriptions, are used as a reference for demonstrating that the hybrid reasoning system performs better than the single reasoning components executed separately. Indeed, it will be shown that by running the hybrid reasoning cycle more than once the system yields better approximation for entities whose extensions are not precisely known.

Since new information was discovered during the first iteration, a new hybrid reasoning iteration is required. The qualitative reasoner infers new relations, one being $Oc(O_4^*, O_2, O_3)$ that the *GeometricReasoner* employs to refine O_4^+ as shown in Fig. 6.7(a). The link $S(O_9^*, O_4^*)$ propagates the refinement to O_9^* as depicted in Fig. 6.7(b), while the descriptions of other entities remain unchanged. The qualification process refines some edges of the constraint network and the whole cycle is run once more. In the last execution, no new information is discovered and the reasoner terminates.

Comparing the resulting objects (Fig. 6.6(b)-(c) respectively for O_5^* and O_6^* , and Fig 6.7 for O_4^* and O_9^*) with the original ones (Fig. 6.5(a)) shows that the reasoner works as intended since any imprecise object contains the original region it represents. Furthermore, the refinements of the imprecise objects O_4^* and O_5^* in the second hybrid reasoning iteration demonstrate that the hybrid reasoner yields (at the quantitative level) better results than the qualitative and geometric approaches would have been able to produce individually. The same can be proven at the qualitative level comparing the relations refined during the reasoner iterations. This results are further confirmed by the fact that multiple reasoning iterations were needed to compute the final descriptions

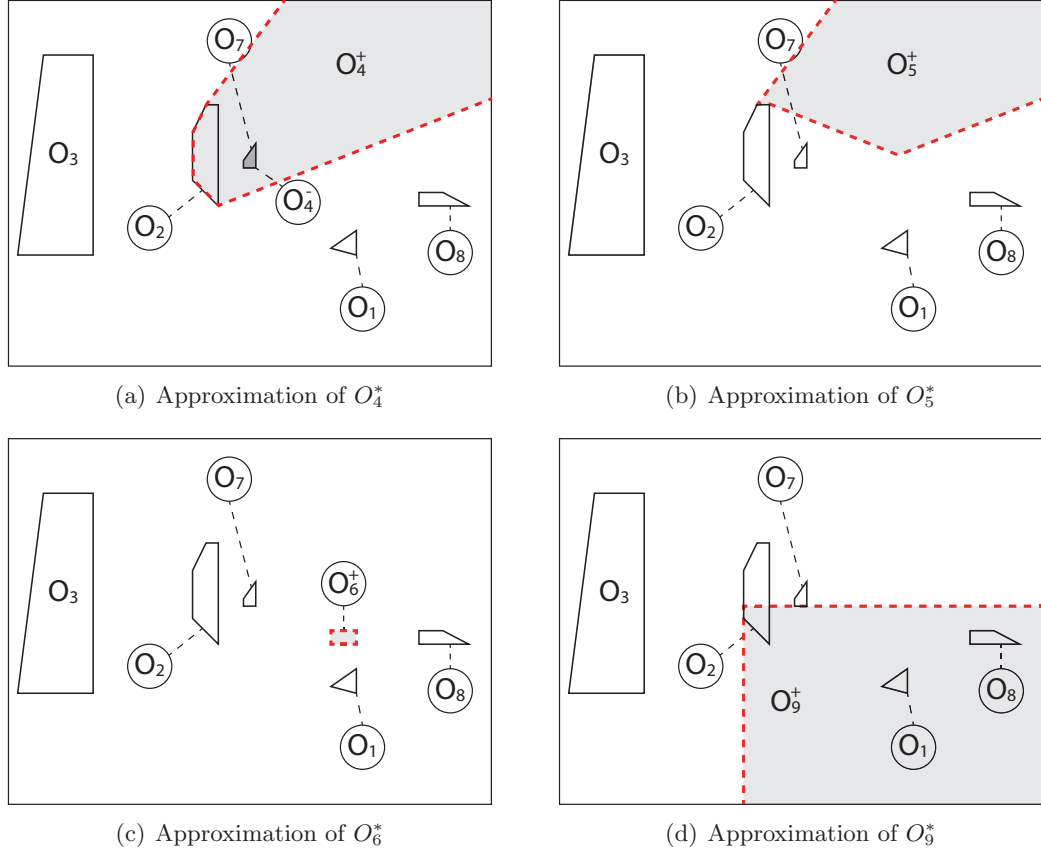


Figure 6.6: First approximations of O_4^* , O_5^* , O_6^* , and O_9^* .

for unknown entities. Indeed, whenever the reasoner requires more than two iterations (considering that the last iteration never produce refinements neither at the qualitative nor at the quantitative level), it means that one single iteration is not enough to refine the information as much as possible.

6.6.2 Heuristics for Reducing the Computation Time

Several heuristics are defined in order to improve the empirical performance of the hybrid reasoning system. Even though they do not affect the theoretical computational time complexity discussed above, they allow for a reduction of the actual amount of information to analyze and hence the required computation time decreases.

QUANTIFICATION OF A QUALITATIVE RELATION.

Let O_i^* be an imprecise object, in the first iteration of the hybrid reasoning procedure all relations whose primary object is O_i^* are quantified. In the next iteration, if the n -ary relation $R(O_i^*, O_2^*, \dots, O_n^*)$ has not been refined and the geometric descriptions of

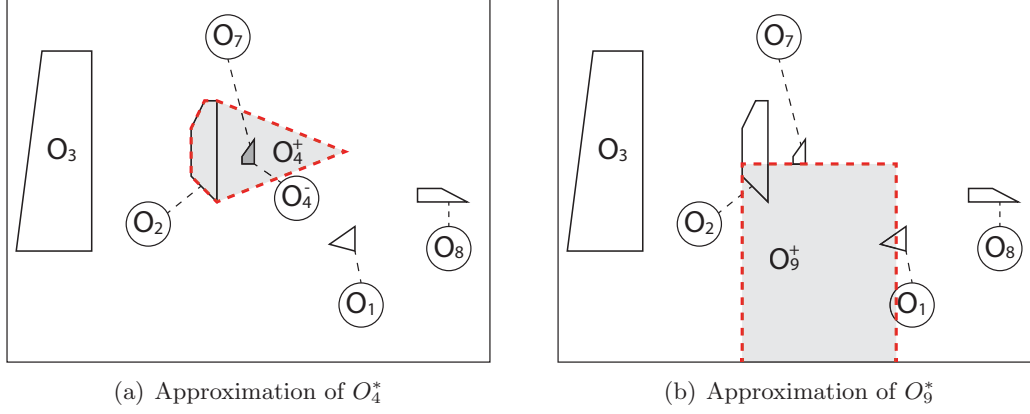


Figure 6.7: Final approximations of O_4^* and O_9^* .

O_2^*, \dots, O_n^* have not been modified, it is not necessary to quantify the relation again. Indeed, the relation has been already used to refine the quantitative description of O_i^* and it would not yield any further approximation. As an example, three spatial-region objects O_1^*, O_2 and O_3 are considered (O_2 and O_3 are precise objects) together with the relations $\{N, NE\}(O_1^*, O_2)$ and $\{SE, S, SW, \}(O_1^*, O_3)$. During the first iteration all relations are quantified for defining the approximation of O_1^* . It is supposed now that the relation between O_1^* and O_2 is refined during the first hybrid reasoning iteration as $N(O_1^*, O_2)$, while the other one is not refined. Thus, in the next iteration only the former relation has to be quantified again.

QUANTIFICATION OF PROJECTIVE DISJUNCTIVE RELATIONS.

Every base relation that composes a disjunctive projective relation (i.e., CDC or Visibility relation) is quantified at a time. For instance, if $\{N:NE, NE:E\}(O_1^*, O_2)$ has to be quantified, the quantification procedure (Algorithm 3, p. 76) quantifies at first $\{N:NE\}(O_1^*, O_2)$, next $\{NE:E\}(O_1^*, O_2)$, and finally performs the union of the results. However, both the quantifications rely on the computation of $A_{NE}^+(O_2)$, that is hence built twice. The performance of the algorithm improves if the quantification procedure quantifies any tile only once. Hence, a modification is made in the algorithm that first computes the union of the tiles that compose all base relations, and afterwards computes the quantification of the relation. In the example, this would result in the quantification of $\{N:NE:E\}(O_1^*, O_2)$.

For the same reason, if the relation to quantify is the universal relation, or if the tile union corresponds to all tiles defined in the projective calculus, the infinite-region representation of \mathbb{R}^2 is directly returned by the quantification procedure.

QUALIFICATION OF RELATIONS BETWEEN PRECISE OBJECTS.

The extension of precise objects is never affected by the quantification procedure.

Indeed, precise objects represent spatial entities for which the quantitative description is already precisely known by the system. Furthermore, the information is consistent and there do not exist relations in which a precise object is the primary object and whose quantification would change its description. Hence, the qualification procedure can consider n -tuple of precise objects only during the first hybrid reasoning iteration, and avoid the qualification between them in the other iterations, resulting in a reduction of the number of permutations to analyze in Algorithm 8.

QUALIFICATION OF RELATIONS BETWEEN IMPRECISE OBJECTS.

For considerations similar to the previous point, the qualification of an n -tuple of imprecise objects is avoided if their approximations did not undergo any change during the previous iteration of the hybrid reasoning cycle.

REDUCE ALGEBRAIC CLOSURE RELATIONS.

For the assumptions defined in Section 3.2.1.1, the system's input information is consistent. This means, for every n -ary calculus there exists only one relation satisfied by an n -tuple of spatial entities. Hence, if the known qualitative relation for a calculus C between an n -tuple is a base relation in the calculus, it is not necessary to propagate other relations on that constraint. Thus, before computing a permutation or a composition (and the correspondent *refineRelation* function), it is checked whether the resulting constraint is a base relation: if it is not, the operation is performed. For instance, if it is known that $DC(O_1^*, O_2^*)$, it is not necessary to compute the composition $R_{O_1^* O_i^*} \diamond R_{O_i^* O_2^*}$.

6.6.3 Combination of Qualitative Calculi

Different approaches for the integration of qualitative spatial calculi have been discussed in Section 2.2.4 (p. 38). The integration is required when it is necessary to deal with several qualitative aspects of the space at the same time. The approaches are clustered into *joint satisfaction problem*, *combined models*, and *combined reasoning*. The first approach (e.g., Gerevini & Renz, 2002; Li, 2007; Liu *et al.*, 2009) is not in the scope of this work, since it focuses on the consistency of relations defined in different calculi¹.

The second approach is grounded on the definition of a new qualitative calculus that merges together the relations defined in calculi that model the single aspects of space at hand (e.g., Billen & Kurata, 2008; Brageul & Guesgen, 2007; Frank, 1992). Even though in this work different aspects of space have been considered, the definition of a new combined model is not necessary. Indeed, the hybrid reasoning system automatically exploits the geometrical properties of the relations defined in any single calculus to infer the relations in other calculi. On the one hand, this approach does not always ensure a

¹The multi-calculus constrain network defined in Section 6.2 is compatible with the *joint network* defined in the *Joint Satisfaction Problem*.

result as precise as the definition of a new model (that expressly excludes the concurrent existence of some relations). On the other hand, the proposed approach simplifies the integration of a new calculus into the system, since only the definitions of the quantification procedures and the reasoning operations have to be done. Furthermore, a combined calculus can be directly included in the system.

In comparison with *combined reasoning* approaches (e.g., Clementini *et al.*, 1997; Guo & Du, 2009; Sharma, 1996)—where rules are defined to perform reasoning algorithms with different calculi at the same time—the hybrid reasoning system does not require an explicit definition of reasoning rules, rather some of them are automatically exploited through the quantification-qualification process. However, *heterogeneous*, *mixed* and *integrated* spatial reasoning rules (Sharma, 1996) can be directly included into the system.

6.7 Thematic Based Reduction

In order to complete the discussion about the integration of geographic information, in this section an improvement of the system described in Section 6.3 is briefly introduced that considers also *thematic information*. Such information is linked to any spatial-region object in G in order to describe the represented entity¹. The symbol \rightsquigarrow is used to assign a label L to a spatial object. For instance, $O_1 \rightsquigarrow \{Building\}$ means that O_1 is a building (e.g., a house or a school), and $O_2 \rightsquigarrow \{WaterBody\}$ means that O_2 is a water body (e.g., a lake or a river). More than one label can be assigned to any object (e.g., $O_1 \rightsquigarrow \{Building, School\}$).

The semantics of thematic information associated to the set of objects is exploited to define *qualitative rules*, that express which qualitative relations are not admissible between two (or more) objects labeled by certain values. Formally, a qualitative rule is a list $\phi = \langle L_1, \dots, L_n, R_\phi \rangle$, where L_1, \dots, L_n are labels and R_ϕ is an n -ary relation that can not hold between the objects that are labeled respectively with L_1, \dots, L_n . As an example, suppose that a constraint is given for objects labeled with *Building* that can not overlap objects labeled as *WaterBody*. This concept can be modeled as the set of qualitative rules (denoted as Φ):

$$\begin{aligned} \phi_1 &= \langle Building, WaterBody, PO \rangle & \phi_2 &= \langle WaterBody, Building, PO \rangle \\ \phi_3 &= \langle Building, WaterBody, EQ \rangle & \phi_4 &= \langle WaterBody, Building, EQ \rangle \\ \phi_5 &= \langle Building, WaterBody, TPP \rangle & \phi_6 &= \langle WaterBody, Building, TPP \rangle \end{aligned}$$

¹In the literature, thematic information is also called *thematic extent* or *conceptual space* occupied by the entity (e.g. Duckham *et al.*, 2006)

The geographic information integration layer is extended by adding a *thematic reduction* component, as Fig. 6.8 shows. Since the thematic information does not undergo any change during the information integration process, it is worthless to include the thematic reduction into the reasoning cycle, rather it runs once before the cycle starts, in order to reduce beforehand the information in the constraint network.

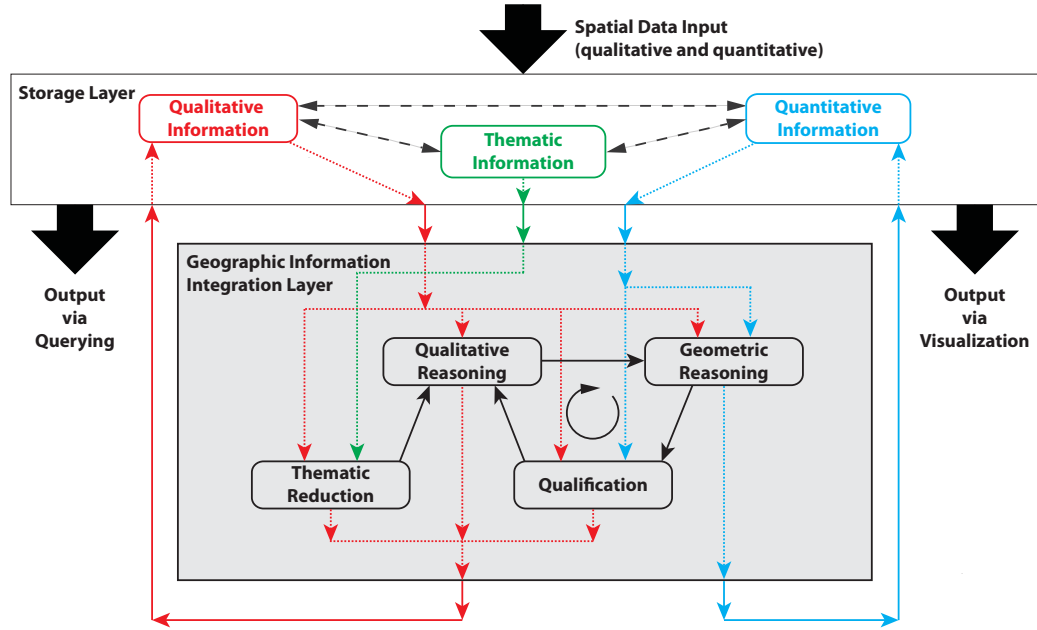


Figure 6.8: Hybrid spatial reasoning system with the thematic reduction component¹.

The procedure that performs the thematic reduction is shown in Algorithm 14. It gets as input the set of spatial-region objects G , a multi-calculus constraint network MN , a list of associations TA between objects and thematic information, and the set of qualitative rules Φ . The algorithm loops through all edges of the constraint network and, for each rule, checks whether the labels associated to the objects are the ones defined in the rule: if so, the relation R_ϕ is removed from the constraints between the given set of objects. The modified constraint network is then returned by the algorithm.

If the number of elements in G (hence the number of nodes of MN) is n , and r is the number of rules defined in Φ , the algorithm runs in $\mathcal{O}(rn^m)$ time in the worst case, where m is the maximal arity of the considered calculi.

¹The red lines in the figure represent the flow of qualitative information: solid lines show the stream of information among the different system's layers, while dotted ones depict how the information flows within any single layer. In the same way, the cyan lines represent the stream of quantitative information and the green lines depict the flow of thematic information. Finally, solid black lines are used to represent the reasoning system's cycle.

Algorithm 14 *ThematicReduction* (G, MN, TA, Φ)

```

for  $R_{O_i^* \dots O_{i+n-1}^*} \in MN$  do
  for  $\phi \in \Phi$  do
    if  $O_i^* = L_1, \dots, O_{i+n-1}^* = L_n$  then
       $R_{O_i^* \dots O_{i+n-1}^*} \leftarrow R_{O_i^* \dots O_{i+n-1}^*} \setminus R_\phi$ 
    end if
  end for
end for
return  $MN$ 

```

The thematic reduction component can be enforced by defining the qualitative rules through a formal language—e.g., *description logics* (Calvanese *et al.*, 2001; Donini *et al.*, 1996). The grounding idea is that if a qualitative rule is defined for a label L_1 , and if it is known that L_2 *is a* L_1 , then the rule applies also to L_2 . For instance, if it is defined that the label *House* *is a* *Building*, and *Lake* *is a* *WaterBody*, then the rules $\phi_1 - \phi_6$ apply also to objects assigned with these labels. Formal languages and conceptual reasoning procedures allow the system to automatically propagate the rules.

6.8 Summary

This chapter described a hybrid spatial reasoning system able to perform spatial inference in mixed setting, where information is given partially *qualitatively* and partially *quantitatively*. The system exploits the quantification and qualification procedures developed in the previous chapters, as well as established inference techniques such as the algebraic closure algorithm for performing qualitative spatial reasoning. It has been shown that the hybrid reasoning system performs better than the single inference strategies executed separately: indeed, it is able to produce refined information, both at the qualitative level (i.e., refined qualitative spatial relations) and at the quantitative level (i.e., better geometric approximations for unknown entities). Furthermore, the system is compatible with existing approaches to combine qualitative spatial information.

Finally, an extension of the system has been proposed that considers also *thematic* information. It is used to refine the qualitative spatial knowledge before the hybrid reasoning cycle starts. Moreover, it opens the way for integrating existing conceptual reasoning techniques into the hybrid reasoning system.

Chapter 7

Experimental Evaluation

In this chapter, an empirical evaluation of the algorithms described in the previous three chapters will be shown. At first, a prototypical implementation of the *geographic information integration system* is described in Section 7.1. The prototype is then used to perform empirical experiments (Section 7.2) conducted for the evaluation of the quantification algorithms (Chapter 4), the qualification algorithms (Chapter 5), and the hybrid reasoning system discussed in Chapter 6.

7.1 Prototype Geographic Information Integration System

The geographic information integration system has been implemented in order to collect experimental results. In this section, the design of the prototype, as defined in Section 6.1, is described. In particular, a solution for spatial data input/output is proposed in Section 7.1.1, the storage layer design is done in Section 7.1.2, and finally the design of the geographic information integration layer is shown in Section 7.1.3.

The current prototype supports RCC-8 relations as well as single and multi-tile relations of CDC and Visibility calculi.

7.1.1 Spatial Data Input, Visualization, and Querying

Quantum GIS (Hugentobler, 2008)—also known as *QGIS*—is a Free Open Source desktop GIS software developed as an easy to use geographic data viewer, later extended to provide functionalities to manage raster and vector data (Steiniger & Bocher, 2009). The option to write extensions in Python to add custom functionalities makes QGIS a good basis for the implementation of the hybrid reasoning system. Fig. 7.1 shows

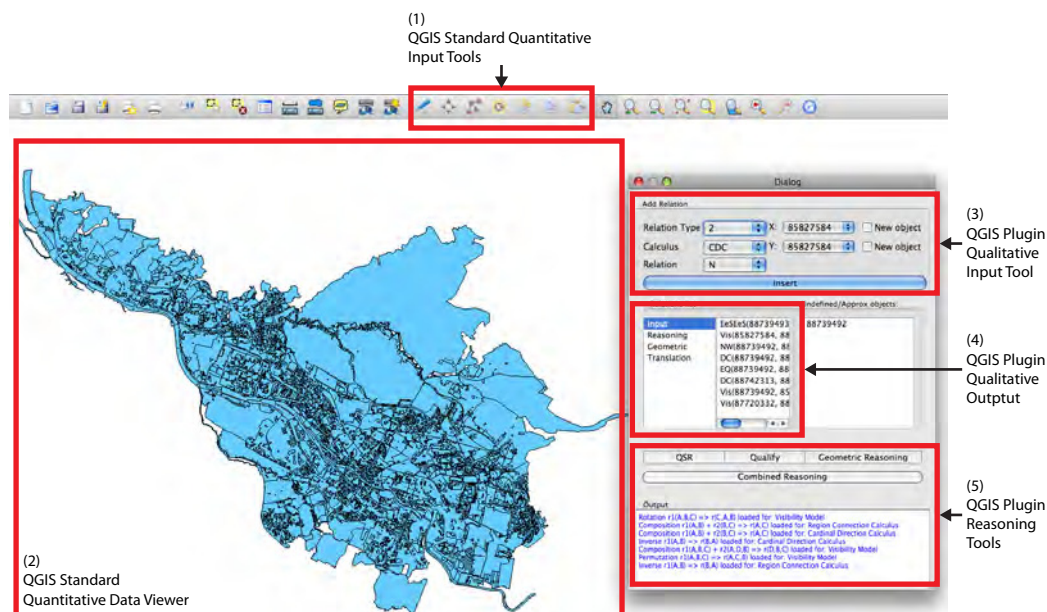


Figure 7.1: QGIS Python plugin for spatial data input/output and for the interaction with the Geographic Information Integration Layer. Input of quantitative data is performed through the QGIS standard input tools (1) that allow for drawing and editing polygons. QGIS also provides instruments for data viewing (2); the example shows the OSM dataset of Bremen. A Python plugin has been embedded in QGIS to provide functionalities for qualitative data input (3) and output (4). The plugin offers tools for executing the reasoning components (5).

the QGIS plugin that has been developed to manage the input and output of qualitative spatial relations. The input and visualization of quantitative data are performed through the QGIS standard user interface.

7.1.2 Storage Layer

As grounding for the storage of quantitative and qualitative spatial information, *PostgreSQL* DBMS¹, with its *PostGIS* spatial extension (Ramsey, 2010), has been chosen due to its capability to store and manage big amounts of both spatial and non-spatial data.

The logical design of the database structure for the storage of spatial information is given in Fig. 7.2. Quantitative description of spatial entities are stored in the *Geometries* table, that maintains two geometries to define the egg and the yolk of imprecise objects (if the *Precise* flag is set to *False*) or one geometry for precise objects (*Precise* flag set to *True*); information for the identification of the objects is stored in the table as well.

¹PostgreSQL: <http://www.postgresql.org/>

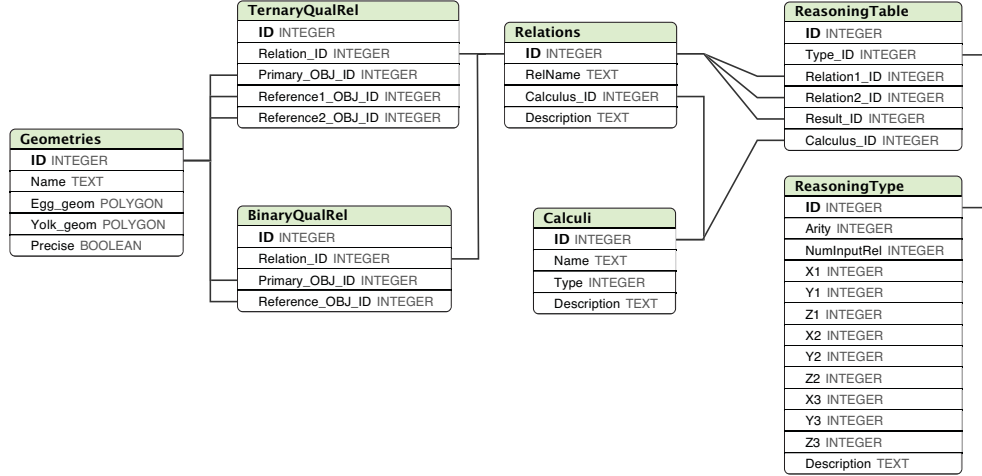


Figure 7.2: Storage layer – *Logical schema*. The table *Geometries* maintains the geometric description of spatial entities; it stores both precise and imprecise objects. The data-structure composed by the *Calculi*, *ReasoningTable*, *ReasoningType*, and *Relations* tables models qualitative relations defined by different calculi as well as reasoning operations. The link among quantitative and qualitative information is maintained by the *TernaryQualRel* and *BinaryQualRel* tables.

A data-structure composed by *Calculi*, *ReasoningTable*, *ReasoningType*, and *Relations* tables has been introduced to model the qualitative relations defined by different calculi as well as their composition and inverse/rotation operations (as generalized in Section 6.5.1). The relations satisfied by pairs (or triples) of objects in the *Geometries* table are stored in *TernaryQualRel* and *BinaryQualRel* that act as connection layer between quantitative and qualitative information.

7.1.3 Geographic Information Integration Layer

The Phyton plugin developed for the input and output of qualitative spatial relations (cf. Section 7.1.1) also implements the geographic information integration layer developed in Chapter 6. A class diagram, representing the core of the system, is drawn in Fig. 7.3. Auxiliary classes, to manage for instance the communication between the plugin and the underlying database, are not detailed in the schema for the sake of readability.

Classes that are depicted with green background model the different kind of spatial information managed by the system: *QualitativeRelation* objects relate *EggYolkObject*¹ objects that are in turn composed by *InfiniteRegion* objects. The class *InfiniteRegion*

¹As shown in Chapter 3.3.3, a precise object is represented as an egg-yolk object having coincident egg and yolk.

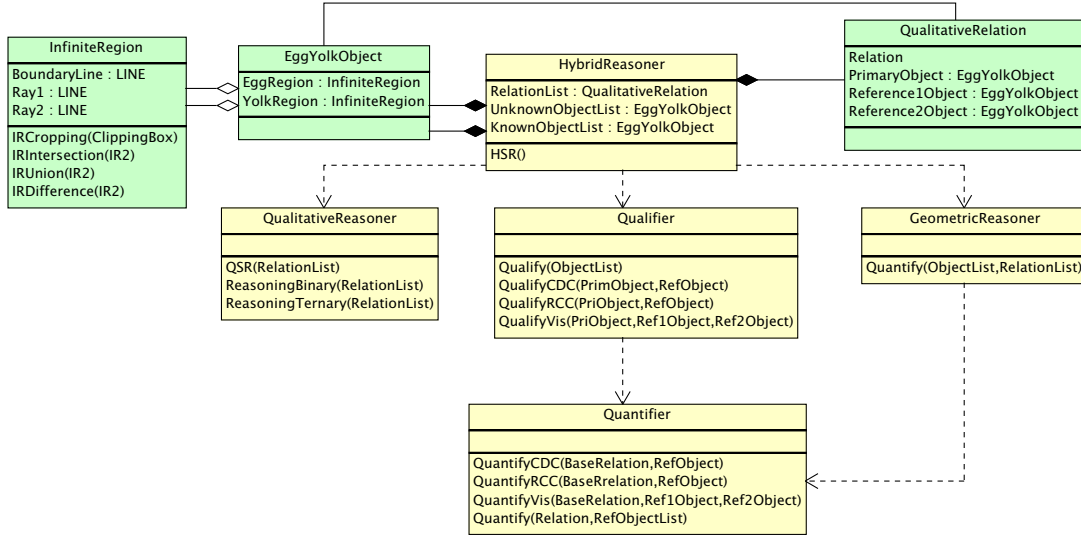


Figure 7.3: Geographic information integration layer – *Class diagram*. Classes depicted with green background model quantitative and qualitative descriptions of spatial knowledge. The spatial entities are quantitatively described as *EggYolkObjects* instances, that are composed by two *InfiniteRegion* objects: one for the egg description and one for the yolk description. *QualitativeRelation* instances model the qualitative relations between spatial objects. The spatial knowledge is used by the classes with yellow background that implement the components of the hybrid reasoning system developed in the previous chapters.

implements the infinite-region object representation (Section 4.3), and the algorithms for computing union, intersection, and difference of infinite-region objects defined in Appendix A. Classes depicted with yellow background model the components of the hybrid spatial reasoning system (Section 6.1) and implement the algorithms developed in the previous three chapters.

7.2 Evaluation of the Geographic Information Integration Layer

The theoretical computational time complexity and the correctness of the spatial information translation procedures as well as the complexity and the correctness of the hybrid spatial reasoning system have been discussed in the previous chapters. The prototype software discussed in Section 7.1 has been used for performing empirical evaluations of the system’s performance: Quantification and qualification components are evaluated only with respect to computation time requirements. Three different measures are instead used for evaluating the hybrid reasoning system: computation time requirements, qualitative output, and quantitative output; moreover, the benefits

of the heuristics for reducing the computation time are shown, and an analysis of the number of hybrid reasoning iterations is done. In all the experiments, the computation time requirements have been measured on a 2.53 GHz Intel Core 2 Duo CPU.

As measure for the goodness of the computation times, two different kinds of interactions with the system are considered within the emergency management context defined in Chapter 3: *background computation* operations and *single-request* operations. Background computation operations require the system to improve the spatial knowledge base as much as possible, both in qualitative and quantitative terms. The integration of qualitative and quantitative spatial knowledge for emergency management, as an alternative to traditional collection methods, is a background computation operation. These operation are performed by the hybrid reasoning system. In order to fulfill the information-demand provision gap (cf. Section 3.1) and to provide emergency responders with update information when needed, the time required for executing background computation operations is acceptable if it does not exceed the time required to collect data through traditional collection methods (cf. Figure 3.2); the computation time is hence acceptable if it is in the order of magnitude of few hours. In contrast, single-request operations are performed when the user directly interacts with the system requesting specific information either at the qualitative or at the quantitative level that needs to be computed at runtime. For instance, a user asks the system to visualize on his map a hazardous area that he knows to be north of his actual position (quantification of a single relation), or the user requests the position of a specific entity with respect to his actual position (qualification of a single relation). Single-request operations are performed by quantification and qualification components only¹. These operations ask for fast human-computer interactions; the system response time is acceptable if it is in the order of magnitude of few seconds.

The OpenStreetMap² (OSM) dataset of Bremen (Fig. 7.4(a))—called *OSM Bremen*—is used as reference for the evaluation of outcomes. It contains 17321 polygons whose number of vertices varies between 3 and 960. Among them, 95% of the objects are defined by 3 to 25 vertices (Fig. 7.4(b)), while 60% of the objects are defined by 4 vertices. The average number of vertices in the whole dataset is 10.

¹Even though the system has been designed for executing background computation operations, quantification and qualification components can be directly executed to perform single-request operations as well.

²OpenStreetMap: <http://www.openstreetmap.org/>

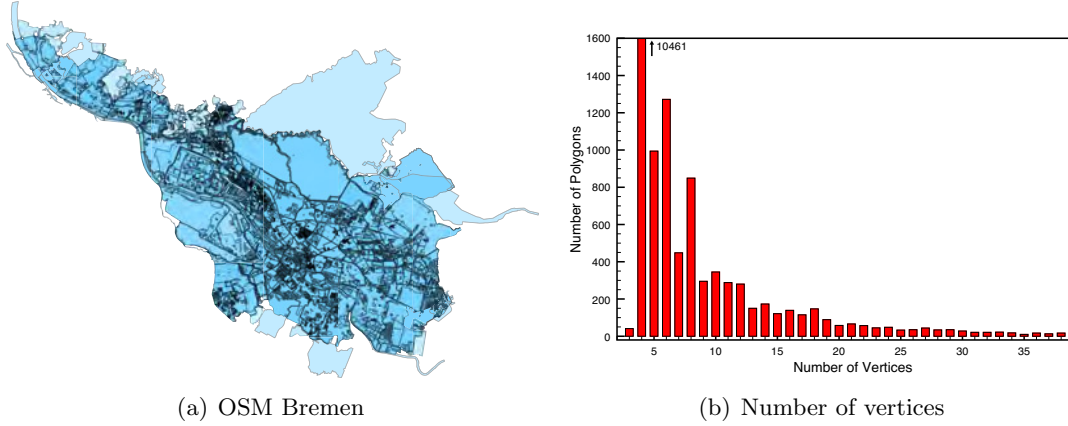


Figure 7.4: OpenStreetMap dataset – Bremen.

7.2.1 The Quantification Component

The aim of the experiments reported in this section is to analyze whether the time necessary for computing the quantification of qualitative spatial relations meets the time constraints for single-request operations. The correctness of the algorithms has been proven in Chapter 4, and therefore it does not require empirical evaluation.

Data in OSM Bremen has been acquired through traditional collection methods, thus it precisely represents the extent of entities that exist in the reality (cf. Section 3.3.3). However, the quantification component often deals with objects which represent spatial regions imprecisely as egg-yolk objects. Indeed, the component is run both for relations whose reference regions are precisely described and for relations whose reference regions' extent has been previously approximated by the component itself (cf. Section 3.3.4). Thus, OSM Bremen is not appropriate for testing the quantification outcomes in all cases. Rather, a dataset that contains both precise and imprecise objects is required.

Performance of quantification depends on the number of vertices (or edges) that define the reference spatial-region objects of the relation (cf. Section 4.7). Hence, the evaluation requires to run the quantification of all relations—both single and multi-tile when applicable—over a set of spatial-region objects while varying the number of their defining vertices. Furthermore, the quantification performance does not depend neither on the spatial distribution of the objects in the dataset¹ nor on the distribution of the points that define any single spatial object. According to these observations, the empirical evaluation of the quantification component can be done over a set of spatial-region objects randomly generated within a defined workspace. Even though the

¹The only constraint is given for the quantification of visibility relations which requires reference objects having non-overlapping convex hull.

objects are randomly generated, the relevant properties that influence the quantification performance are preserved with respect to real datasets (e.g., OSM Bremen for precise objects).

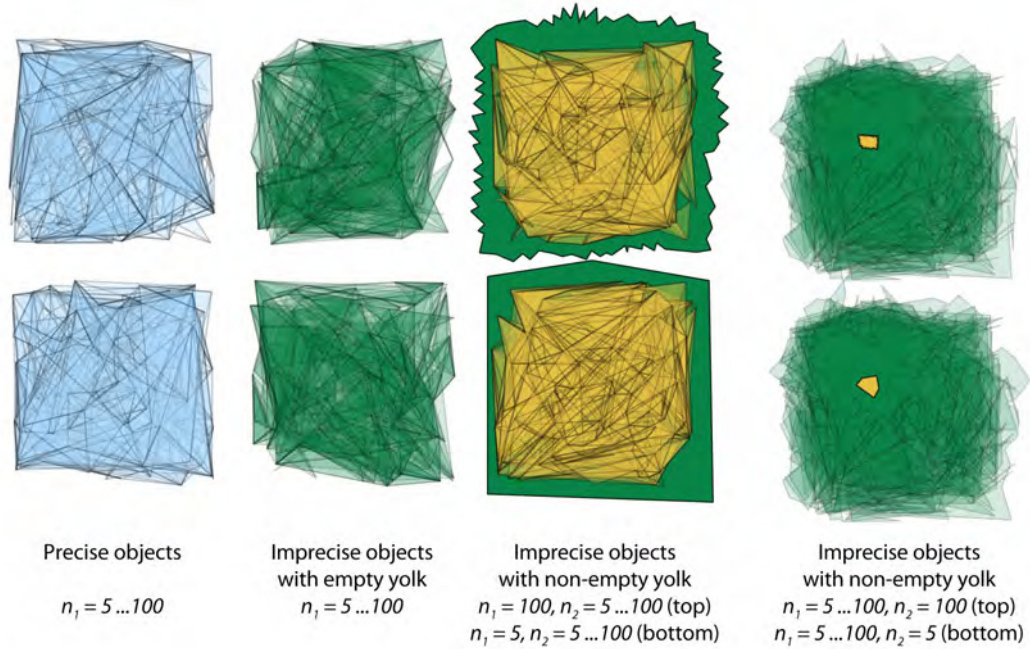


Figure 7.5: Quantification testbed. The objects have been randomly generated. Blue regions depict precise objects, green regions represent eggs of imprecise objects, and yellow regions stand for yolks of imprecise objects. n_1 is the number of vertices that define precise objects or the egg of imprecise objects, while n_2 is the number of vertices that define the yolk of imprecise objects.

The *quantification testbed* has been randomly generated that contains 160 distinct spatial-region objects; a tool¹, grounded on Zhu *et al.* (1996), has been used for generating random simple closed polygon with n vertices in a given workspace. In order to meet the requirements for the computation of visibility relations' quantifications, the workspace has been split into eight non-overlapping cells, each of which contains 20 spatial-region objects of the same type (either precise or imprecise with empty yolk or imprecise with non-empty yolk). The quantification testbed is composed by:

- 40 precise objects described by single polygons having n_1 vertices, with $n_1 \in \{5i \mid i = 1 \dots 20\}$;
- 40 imprecise objects with empty yolk described by single polygons having n_1 vertices, with $n_1 \in \{5i \mid i = 1 \dots 20\}$;

¹<http://caschi.org/downloads/windows-binaries/genpoly.zip>

- 20 imprecise objects with non-empty yolk each composed by two polygons, representing respectively the egg and the yolk, having n_1 and n_2 vertices, with $n_1 \in \{5i \mid i = 1 \dots 20\}$ and $n_2 = 5$;
- 20 imprecise objects with non-empty yolk each composed by two polygons, representing respectively the egg and the yolk, having n_1 and n_2 vertices, with $n_1 \in \{5i \mid i = 1 \dots 20\}$ and $n_2 = 100$;
- 20 imprecise objects with non-empty yolk each composed by two polygons, representing respectively the egg and the yolk, having n_1 and n_2 vertices, with $n_1 = 5$ and $n_2 \in \{5i \mid i = 1 \dots 20\}$;
- 20 imprecise objects with non-empty yolk each composed by two polygons, representing respectively the egg and the yolk, having n_1 and n_2 vertices, with $n_1 = 100$ and $n_2 \in \{5i \mid i = 1 \dots 20\}$.

The quantification testbed is shown in Fig. 7.5; blue regions depict precise objects, green regions represent eggs of imprecise objects, and yellow regions stand for yolks of imprecise objects.

The empirical results obtained for the quantification of RCC-8, CDC, and Visibility relations over the quantification testbed are summarized in the remainder of this section.

7.2.1.1 Quantification of Topological Relations

In this experiment, the quantification of every topological relation defined in the RCC-8 calculus has been computed once for all spatial-region objects in the quantification testbed. Average (AVG), maximum (Max), and minimum (Min) computation time for computing RCC-8 quantifications are summarized in Fig. 7.6. The results are plotted separately for precise and imprecise objects (either with or without empty yolk). For every value of polygon vertices, the average time is calculated over 16 quantification runs in the case of either precise objects or imprecise objects with empty yolk, and over 32 quantification runs for imprecise objects with non-empty yolk.

The average computation time slightly increases with the number of vertices in all cases; however, the time is lower than $3ms$. Even though the maximum computation time can rise up to five times the average time, it rarely exceeds $10ms$. The computation time of RCC-8 quantifications is independent of the number of vertices that define the reference object of the relation. This results confirm the theoretical time complexity evaluation done in Section 4.7.3.

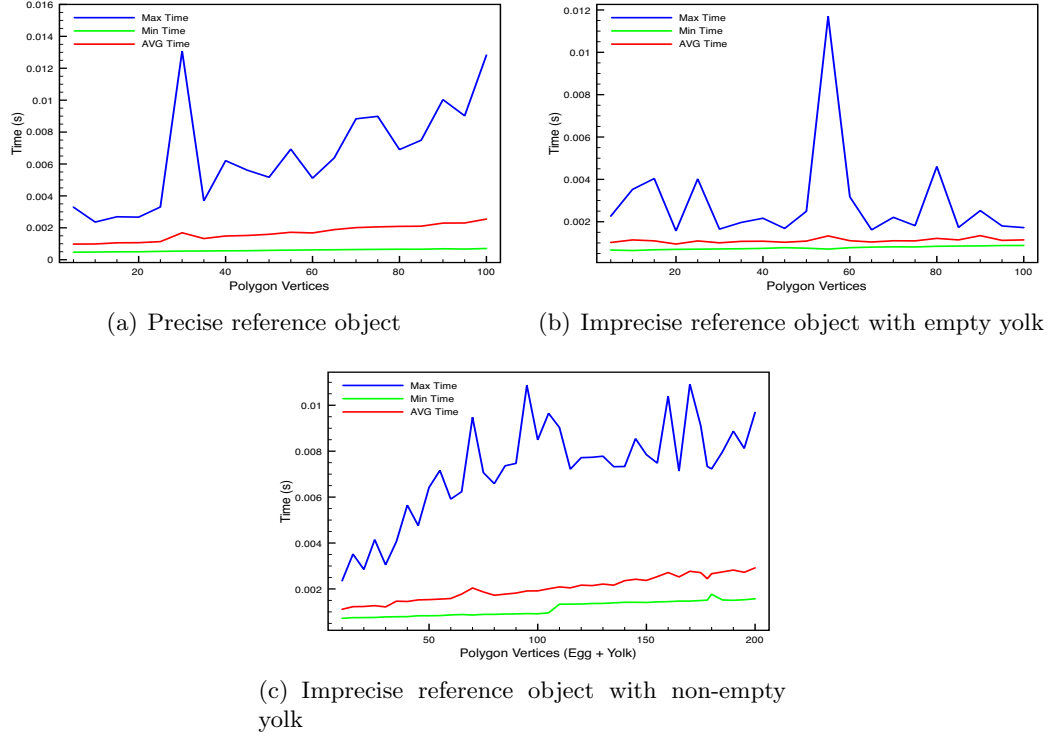
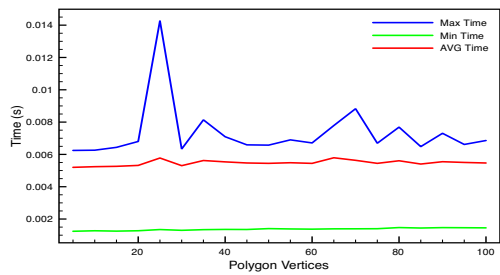


Figure 7.6: Quantification of RCC relations.

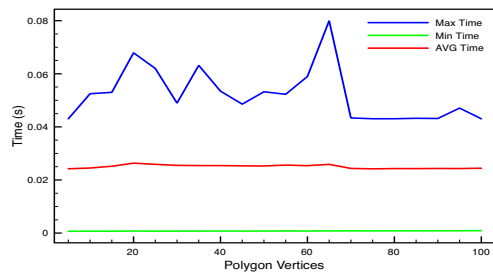
7.2.1.2 Quantification of Cardinal Direction Relations

For any spatial-region object in the quantification testbed, the quantifications of the 218 CDC base relations (both single and multi-tile) have been computed. Average (AVG), maximum (Max) and minimum (Min) computation time for computing the quantifications are plotted in Fig. 7.7 for precise reference objects, in Fig. 7.8 for imprecise reference objects with empty yolk, and in Fig. 7.9 for imprecise reference objects with non-empty yolk. In the first two cases, the results are further distinguished between single and multi-tile relations. For any value of polygon vertices, the average time is calculated over 18 quantifications in the case of single-tile relations and over 418 quantifications in the case of multi-tile relations.

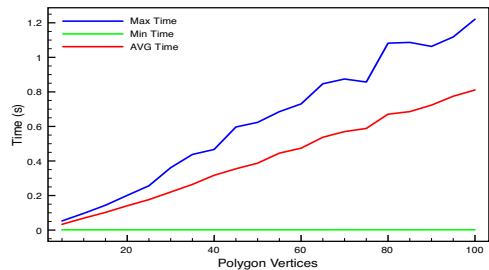
Considering the case if the reference object is precise (Fig. 7.7), the time required for the computation of the quantification of a CDC base relation is practically independent of the number of vertices that define the reference object. Even though the theoretical computation time increases linearly with the number of vertices (cf. Section 4.7.1), for the objects considered in the quantification testbed the time difference is negligible. Indeed, the quantification with a precise reference object only requires a single scan of the object defining vertices in order to identify the maximum and mini-



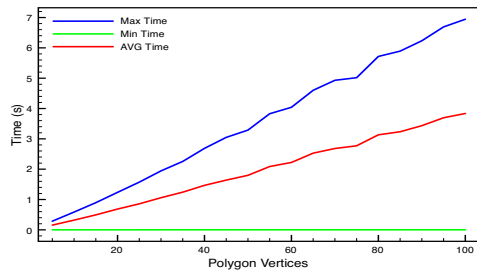
(a) Single-tilde CDC relations



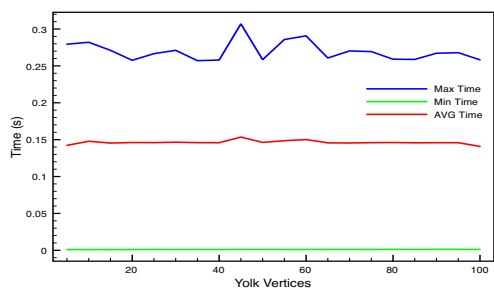
(b) Multi-tilde CDC relations

Figure 7.7: CDC quantification: precise reference object.

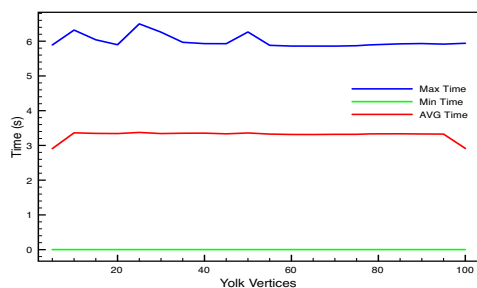
(a) Single-tilde CDC relations



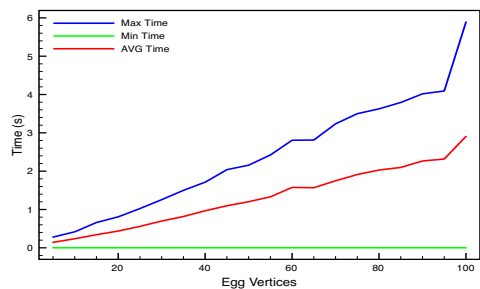
(b) Multi-tilde CDC relations

Figure 7.8: CDC quantification: imprecise reference object with empty yolk.

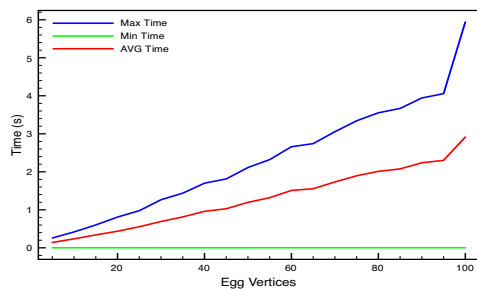
(a) Egg defined by 5 vertices



(b) Egg defined by 100 vertices



(c) Yolk defined by 5 vertices



(d) Yolk defined by 100 vertices

Figure 7.9: CDC quantification: imprecise reference object with non-empty yolk.

imum coordinates values. Furthermore, the average time required for the quantification of single-tile relations (Fig. 7.7(a)) is around five times¹ lower than the average time required for multi-tile relations (Fig. 7.7(b)). This result is motivated by the fact that up to nine different objects are computed and combined for quantifying multi-tile relations. Finally, the maximum time required to compute the CDC quantifications over precise objects rarely exceeds 0.06s.

In the case if the reference object is imprecise with empty yolk (Fig. 7.8), the time increases significantly with the number of vertices that define the reference spatial-region object. Indeed, the quantification in this case requires to compute an object for any edge that defines the reference object, and then to combine all results by union operations. The theoretical computation time—that increases exponentially with the number of vertices—and the strong dependency on the number of vertices is confirmed by the empirical evaluation. As in the previous case, the computation time for the quantification of multi-tile relations (Fig. 7.8(b)) is, on average, five time bigger than the single-tile relations case (Fig. 7.8(a)) due to the bigger number of quantifications to compute and combine.

Finally, if the reference object is imprecise with non-empty yolk (Fig. 7.9), the computation time is independent of the number of vertices that define the yolk, while it increases with the number of vertices that define the egg. Again, the empirical evaluation confirms the theoretical analysis done in Section 4.7.1, where for the imprecise reference object case the computation time increases linearly with the number of yolk vertices and exponentially with the number of egg vertices. The higher average computation time required in the case depicted in Fig. 7.9(b) with respect to the case in Fig. 7.9(a) is caused by the difference between the number of vertices that define the egg; the time is however consistent with the results in Fig. 7.8(b). Conversely, Fig. 7.9(c) and Fig. 7.9(d) show the same performance independently of the number of vertices that define the yolk of the reference object.

7.2.1.3 Quantification of Visibility Relations

Fig. 7.10 and Fig. 7.11 summarize the computation time required for the quantification of visibility relations over pairs of objects belonging to the quantification testbed. For any pair of objects in the testbed with non overlapping convex-hull the quantifications of the 27 visibility base relations (single and multi-tile) have been computed. Fig. 7.10 plots the maximum (Max), minimum (Min), and average (AVG) computation time if both the reference objects are precise, while Fig. 7.11 depicts the case in which at least one of the reference spatial-region objects is imprecise. The parameter n denotes the number of vertices of the first reference object—representing the *obstacle* in the

¹Note that the average number of tiles that compose the multi-tile CDC base relations is 5.12.

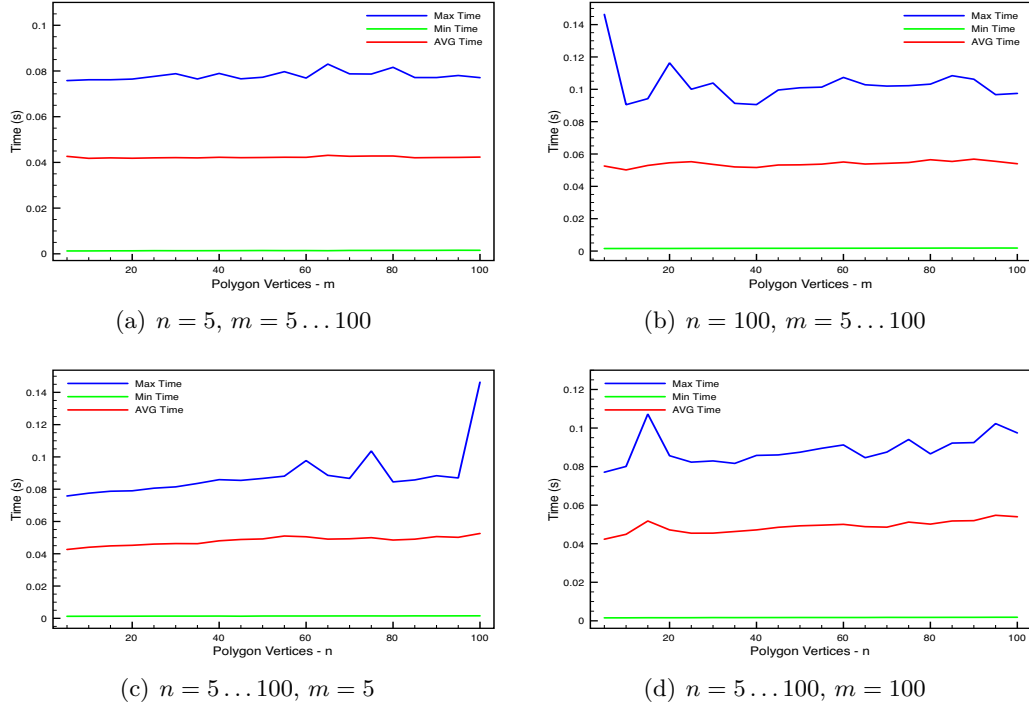


Figure 7.10: Quantification of Visibility relations: precise reference objects.

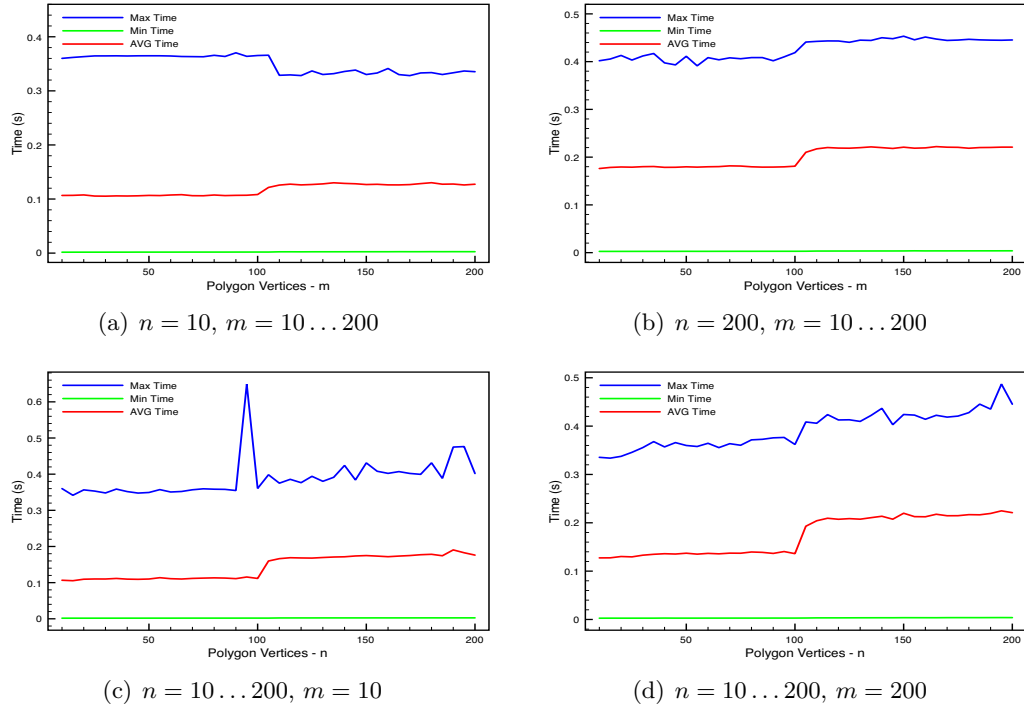


Figure 7.11: Quantification of Visibility relations: imprecise reference objects.

relation—while m denotes the number of vertices of the second object—the *viewer*¹. In order to better emphasize the dependencies of the computation time on the number of vertices that define either the obstacle or the viewer, the results have been further split by keeping fixed the number of vertices of one object while varying the number of vertices of the other object.

All results, both in the precise and in the imprecise case, show that the time necessary for the computation of visibility quantifications is almost independent of the number of vertices that define the viewer object, while it slightly increases with the number of the obstacle defining vertices. The results empirically confirm the theoretical analysis done in Section 4.7.2, where it has been shown that the computation time increases linearly with m and quadratically with n in the worst case. The average time is lower than 0.25s, while the maximum computation time rarely exceeds 0.5s.

7.2.1.4 Discussion

In the experiments described above, the time requirements for computing the quantification of a single qualitative relation have been analyzed for relations in the RCC-8, CDC and visibility calculi. While the times required for computing the quantification of both RCC-8 and visibility relations show negligible dependency on the number of reference objects' vertices—it is acceptable for answering single-request operations (the maximum is reached by visibility relations that require up to 0.6s)—the same does not always hold for CDC relations. Indeed, if the reference object of the relation is imprecise, the quantification strongly depends on the number of defining vertices and the required time doubles when the number of vertices doubles.

The time required to quantify a single CDC relation in the case the reference object is imprecise is close to the time required for quantifying RCC-8 and Visibility relations only if the number of vertices that define the maximal extension of the spatial region is small. In contrast, for objects with a high number of egg vertices the computation time sensibly increases compared with the other cases. However, assuming that the number of vertices that define the egg corresponds to the average number of vertices in OSM Bremen (10 vertices) the time required for the computation is equals to 0.32s, while for polygons having 25 vertices² the time rises to 0.85s that has the same order of magnitude of the quantification of relations defined in the other calculi.

Thus, the quantification of a single qualitative relation over real spatial descriptions has acceptable computation time requirements for single-request operations. However, this result can not be directly propagated to quantifications as used in the qualification

¹In the case of imprecise objects, n and m are the sum of the number of vertices that describe the egg and the yolk.

²95% of elements in OSM Bremen have number of vertices ranging between 3 to 25.

components as well as in the hybrid reasoning system; indeed, in these cases the quantification of more qualitative relations at a time is required. Empirical experiments will be shown for these cases in the next sections.

7.2.2 The Qualification Component

The experiments described in this section aim at evaluating the empirical computation time required for qualifying spatial crisp relations holding between tuples of spatial-region objects (Chapter 5). In particular, it is analyzed whether the computation time meets the requirements of single-request operations. Computation time of crisp relations between spatial-region objects generally depends on three parameters: (1) computation of the quantifications of qualitative relations¹, (2) distribution of the spatial-region objects over the workspace², and (3) properties of spatial-region objects at hand with respect to precision.

The dependency on the first parameter is excluded from the evaluation since it produces the same results shown in the previous section. Furthermore, with respect to the second parameter, entities from OSM Bremen are chosen for testing qualification in order to analyze the performance considering real instances of quantitative spatial descriptions. Thus, in this section only the dependency of qualification on the properties of spatial-region objects (precision) is discussed. In order to meet all the requirements, the *qualification testbed* has been build by selecting 90 polygons from OSM Bremen that describe buildings in the Findorff district. To simulate imprecise descriptions, the selected geometries have been modified as follows:

- 30 polygons have not been modified and describe precise objects;
- 30 polygons have been expanded through a buffering operation (Felkel & Obdrzalek, 1998) and they describe imprecise objects with empty yolk;
- 30 polygons have been modified in order to describe imprecise objects with non-empty yolk. The components of any spatial-regions object have been defined as: yolk equals to the original polygon; egg equals to an inflation of the original polygon obtained trough a buffering operation.

The qualification testbed is depicted in Fig. 7.12. Blue regions represent precise objects, green regions represent the eggs of imprecise objects, and yellow regions represent the yolks of imprecise objects.

In the remainder of this section, the qualification testbed is used as a base for the empirical evaluation of RCC-8, CDC, and Visibility relations' qualification. The symbol

¹This is valid for projective calculi (i.e., CDC and visibility calculus) but it does not apply for RCC-8 relations.

²Indeed, the distribution influences the probability that a certain relation holds between a tuple of spatial-region objects.

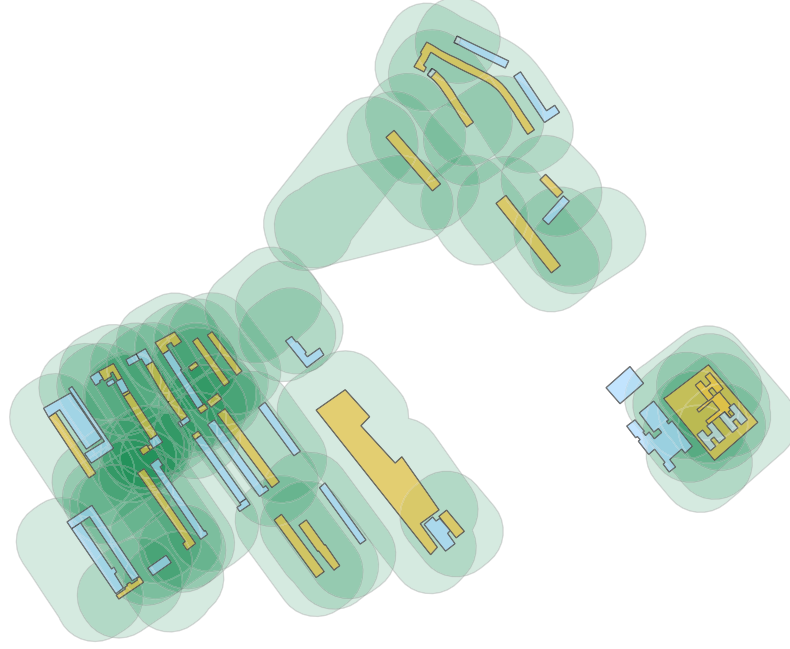


Figure 7.12: Qualification testbed. The objects have been selected from OSM Bremen and opportunely modified to simulate imprecise descriptions¹. Blue regions represent precise objects, green regions represent the eggs of imprecise objects, and yellow regions represent the yolks of imprecise objects.

P is used to denote precise objects, E stands for imprecise objects with empty yolk, and EY refers to imprecise objects with non-empty yolk.

7.2.2.1 Qualification of Topological Relations

In this experiment, the topological relation holding between any pair of distinct spatial-region objects in the qualification testbed has been computed. The maximum (Max), minimum (Min), and average (AVG) computation time are plotted in Fig. 7.13. In order to highlight dependencies on the properties of the objects, the results are aggregated with respect to the characteristics of the objects in terms of precision. The average time for any value in the chart has been calculated over 890 qualification executions.

The average time to qualify topological relations is almost constant for every combination of object types, and it is lower than 0.005s. Computation time slightly increases if imprecise objects are considered, due to more computations required for retrieving the disjunctive relation (cf. Section 5.2). In conclusion, even though in some execution the maximum computation time is 5 times greater than the average time, the computation of RCC-8 relations is always acceptable for single-request operations.

¹Additional information about the testbed is given in Appendix B.

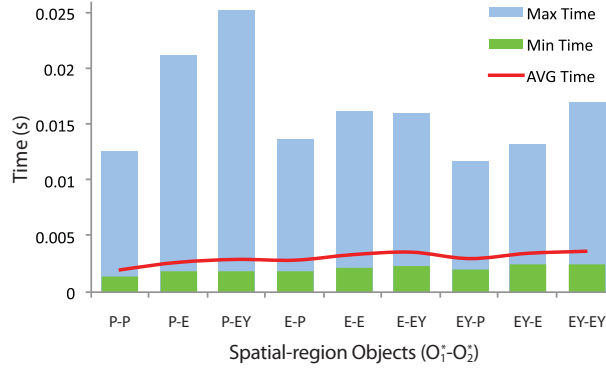


Figure 7.13: Qualification of RCC-8 relations. Any bar in the graph shows the result aggregated based on the object types pair: P denotes precise objects, E stands for imprecise objects with empty yolk, and EY refers to imprecise objects with non-empty yolk.

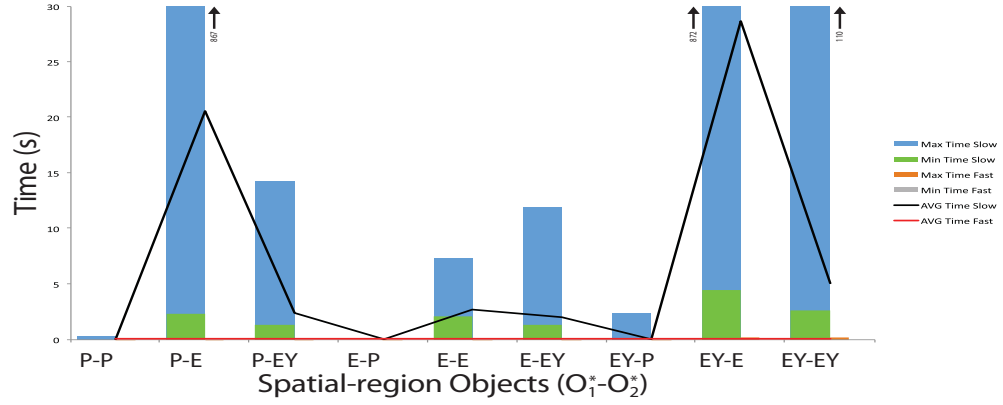
7.2.2.2 Qualification of Cardinal Direction Relations

In Section 5.4, two different approaches have been shown to compute the crisp disjunctive cardinal direction relation holding between two spatial-region objects (either precise or imprecise). The first approach is based on the computation of a considerably high number of quantifications of CDC relations over the reference object. Being the quantification of CDC relations computationally expensive (cf. Section 4.7.1), a different approach has been proposed that exploits the characteristics of the calculus for reducing the number of required computations. In order to empirically prove the benefits of adopting the second approach, qualification tests have been performed executing both the proposed methods.

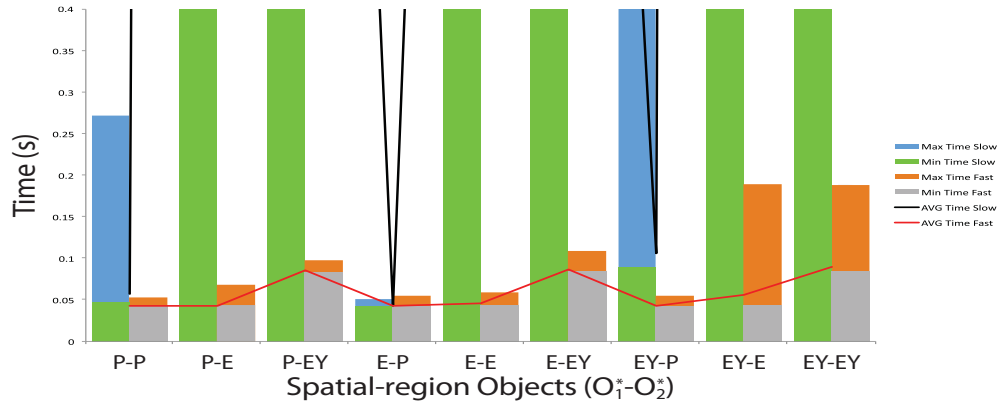
The CDC relation has been computed twice for any pair of objects in the qualification testbed¹, and the computation times have been recorded. The minimum (Min), maximum (Max), and average (AVG) computation time—plotted separately for precise and imprecise objects—are shown in Fig. 7.14(a). For any combination of spatial-region object types, the average time has been calculated over 890 qualification runs. Fig. 7.14(b) plots an highlight of Fig. 7.14(a), where the y-axis scale has been reduced from 30s to 0.4s. *Time Slow* denotes the time required for the computation of the relation with the first approach, whilst *Time Fast* indicates the time required for the computation if the second approach is used instead.

In case the first approach is used, the required computation time is noticeably high. Indeed, in the worst case it is required to compute the quantifications of up to 218 different relations. The average computation time rises up to 30s if both objects are imprecise, and there exist cases where the computation time rises up to 900s. In case

¹For any pair of objects, the relation has been computed both with Propositions 5.4.1-5.4.3 and with the theory developed in Section 5.4.2.



(a) Qualification time



(b) Highlight of Fig. 7.14(a)

Figure 7.14: Qualification of CDC relations. *Time slow* is the computation time for the qualification approach defined in Section 5.4; *Time fast* is the computation time related to the qualification method in Section 5.4.2. Any bar in the graph shows the results aggregated based on the object types pair: *P* denotes precise objects, *E* stands for imprecise objects with empty yolk, and *EY* refers to imprecise objects with non-empty yolk.

the primary object is imprecise with empty yolk, CDC qualification requires, in general, less time than other cases. Indeed, in order to identify the crisp disjunctive relation it is necessary to compute only the quantifications of the cardinal direction single-tile relations (see Section 5.4). However, the required amount of time is, in most cases, not acceptable for single-request operations.

However, the adoption of the second approach drastically reduces the computation time. The computation of CDC relations requires, on average, around 0.1s, that is acceptable for single-request operations. The only drawback in the usage of this approach is that it does not always yield results as precise as the first approach does, as discussed in Section 5.4.3. This event occurs in 10% of the performed tests.

In conclusion, the second approach is preferable for computing CDC relations between spatial-region objects in those cases where less precision in the output is acceptable if the computation time stays low. In contrast, the first proposed approach is desirable in those applications where precise output is required and where no time constraints exist.

7.2.2.3 Qualification of Visibility Relations

In this experiment, the times required for computing qualifications of visibility relations between triples of spatial-region objects in the qualification testbed have been recorded. The average (AVG), maximum (Max), and minimum (Min) computation time are plotted in Fig 7.15, where the results have been aggregated with respect to the types of spatial-region objects involved in the qualification. For any triple of object types, the average computation time has been calculated over 12400 qualification runs.

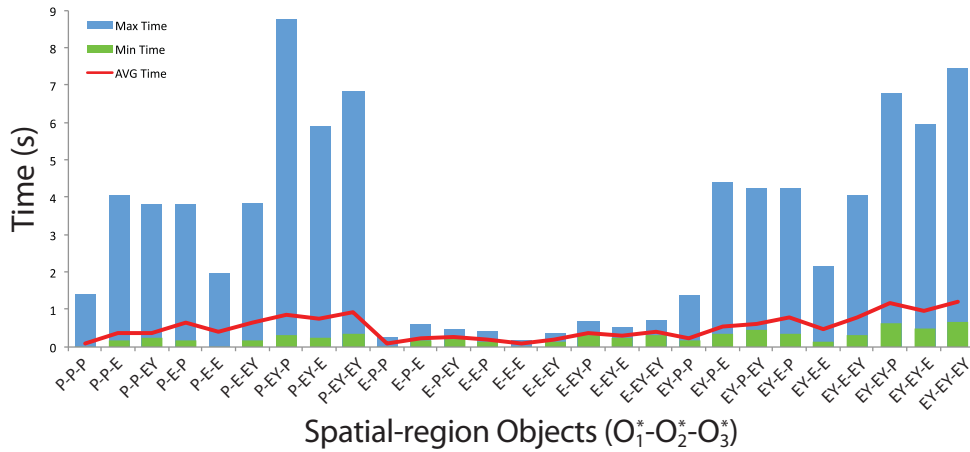


Figure 7.15: Qualification of Visibility relations. Any bar in the graph shows the result aggregated based on the object types triple: P denotes precise objects, E stands for imprecise objects with empty yolk, and EY refers to imprecise objects with non-empty yolk.

The average computation time strictly depends on the properties of spatial-region objects: it increases if imprecise objects with non-empty yolks belong to the triple of spatial-region objects because more computations are required for retrieving the visibility relation (cf. Section 5.3). However, the average time is lower than 1.5s. The maximum time, conversely, rises up to nine times the average computation time: this happens in the worst case where all 27 visibility base relations have to be quantified. In case the primary object is imprecise with empty yolk, even the maximum time is considerably lower than in other cases, because only the quantifications of the five single-tile visibility relations have to be computed to qualify the disjunctive visibility

relation. To conclude, the experiment shows that the time required for qualifying visibility relations between a triple of spatial-region objects is acceptable for single-request operations.

7.2.2.4 Discussion

The experiments shown above aim at empirically analyzing the time requirements of the algorithms developed for qualifying qualitative spatial relations between spatial-region objects. RCC-8, CDC, and visibility calculi have been considered.

Qualification of RCC-8 relations exploits the topological properties of the egg-yolk approach for dealing with imprecise descriptions; few computations are required for qualifying the topological relation. Conversely, the qualification of visibility and CDC relations exploits the properties of the quantification in order to compute the crisp relation between a tuple of spatial-region objects; this method can be directly applied to other projective calculi. However, while the qualification of visibility relations is efficient for single-request operations, CDC qualification required the development of a more efficient approach that exploits the calculus structure. This means, even though the approach developed for qualifying visibility relations—and then adopted for CDC relations—is general enough to be directly adapted to other calculi, the requirements with respect to computation time are not always acceptable for single-request operations.

In conclusion, all results show that the developed procedures require acceptable computation time in terms of single-request operations for qualifying the relation between a single tuple of spatial-region objects. However, those applications where relations between more than one tuple of objects need to be qualified require a specific analysis of the amount of qualifications to compute, as it will be shown for the Hybrid Spatial Reasoning System experiments described in the remainder of this chapter.

7.2.3 The Hybrid Spatial Reasoning System

The capabilities of the hybrid spatial reasoning system developed in Chapter 6 have been evaluated running the system over problem instances generated from the configuration of objects, extracted from OSM Bremen (all belonging to the Findorff district), depicted in Fig. 7.16. It was decided to keep the set of objects small in order to prevent the set of qualitative spatial relations from excessively increasing in size¹, and hence to limit the computation time required by the system. Indeed, as shown in the previous sections, the time requirements for computing a single relation quantification as well as

¹It has to be considered that, for instance, a dataset of 6 geometries yields up to 288 qualitative relations (6^2 RCC-8 relations, 6^2 CDC relations, and 6^3 visibility relations), a dataset of 10 elements yields 1200 relations, and a dataset of 20 elements yields up to 8800 relations.

the qualification of the qualitative relation between a single tuple of spatial-region objects is acceptable for single-request operations; a different evaluation is necessary when more qualifications and quantifications have to be computed, as for the hybrid spatial reasoning system (background computation operation). Hence, it has been decided to limit the computation time and to focus rather on evaluating the dependencies of the system outcome on the number of unknown entities (that has been varied between 1 and 5) and on the number of qualitative input relations (that has been varied between 0 and 15).



Figure 7.16: Hybrid spatial reasoning testbed extracted from OSM Bremen¹.

1100 random input sets—denoted as *HSR input sets* in the remainder of this chapter—were generated by randomly selecting known entities that would be given geometrically and unknown entities only described using qualitative relations. The provided relations involving the unknown entities as either primary or reference entity were also randomly chosen. Computation times, inferred qualitative relations, and geometric descriptions produced for all unknown entities have been recorded for the hybrid reasoning system execution with each of the HSR input sets.

At first, the benefits on the hybrid reasoning performance deriving from the heuristics described in Chapter 6 are empirically analyzed (Section 7.2.3.1). Afterwards, three different measures are used for the analysis of the hybrid reasoning system’s characteristics: (1) computation time (Section 7.2.3.2), (2) number of inferred base and disjunctive (non-universal) relations (Section 7.2.3.3), and (3) quantification of the unknown input entities (Section 7.2.3.4). Finally, an analysis on the number of iterations performed by the reasoning system (Section 7.2.3.5) is done. The outcomes of the hybrid reasoning system (that will be denoted as *HR* in the remainder of this section) are compared with the result obtained by applying the geometric reasoning directly to the input relations (denoted as *GR*), and with the results obtained by executing qualitative spatial reasoning on the input relation set and then performing geometric reasoning (denoted as *QSR-GR*); this is used as a reference for evaluating the benefits

¹Additional information about the testbed is given in Appendix B.

of using the hybrid reasoning system rather than running the single approaches for qualitative and geometric reasoning separately¹.

7.2.3.1 Heuristics

At first, the benefits deriving from the heuristic described in Section 6.6.2 are empirically proven. The heuristics have been introduced in order to reduce the actual computation time, while the theoretical time stays unvaried. HR has been run 35 times with random HSR input sets, both without heuristics (*Heuristics OFF*) and with heuristics (*Heuristics ON*). The time required for the execution of any single HR component during each iteration has been recorded.

The average computation time is reported in Fig 7.17; results are plotted separately for the first reasoning iteration and the other iterations. Indeed, most of the heuristics are active only starting from the second reasoning iteration.

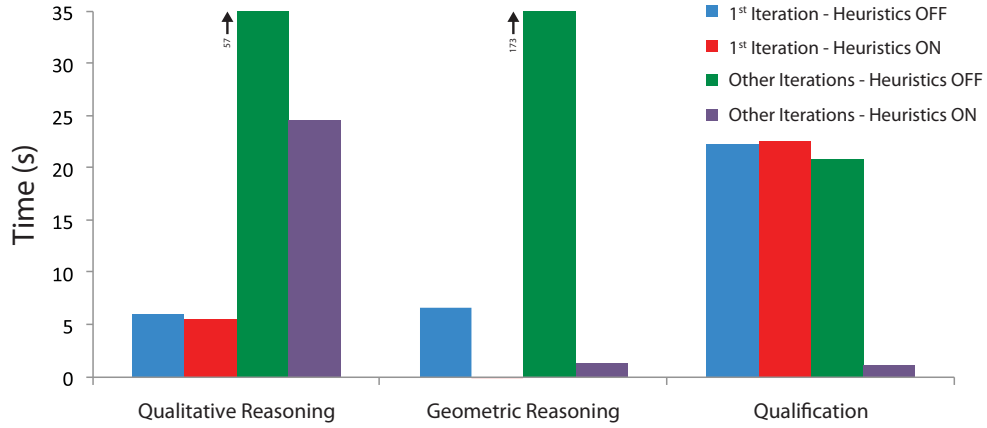


Figure 7.17: Hybrid reasoning system – Heuristics. The graph shows the reduction of the computation time in the case the system implements heuristics for the reduction of the empirical time. Starting from the second reasoning iteration, the system performs several orders of magnitude better than the system without heuristics.

During the first reasoning iteration, only the geometric reasoning component performs better in the case the heuristics are active, since other HR components do not implement heuristics that affect this iteration. Differently, during other HR's iterations all components show a high computation time reduction: The qualitative reasoning component requires 42,5% of the time required by the system executed without heuristics; Geometric reasoning with heuristics requires only 0.7% of the time without heuristics; The qualification component, in the case heuristics are active, employs 5% of the

¹The same reference has been used in Section 6.6.1 for showing the HR benefits with respect to an exemplary input set of spatial-region objects and qualitative relations.

time required in the case heuristics are not active. Overall, the defined heuristics allow for getting the same response from the system in terms of qualitative and quantitative output, reducing the time requirements to 25% of the time required by the system without heuristics.

In conclusion, the proposed heuristics sensibly improve the hybrid reasoning system performance in terms of computation time. The experiments reported in the remainder of this chapter are related to the case heuristics are active in the system.

7.2.3.2 Computation Time

Fig. 7.18 shows how the computation time is affected by the number of unknown entities (Fig. 7.18(a)) and by the number of input relations (Fig. 7.18(b)). Computation times for GR and QSR-GR are not affected by the variation of unknown entities. Indeed, the two components only perform operations over the input relations, hence their computation times only depend on the number of input relations. Computation time for HR does not show dependencies on the number of input relations. In contrast, the computation time of HR sensibly increases with the number of unknown entities. The dependency is caused by the qualification component. As an example, consider the two cases where the number of unknown entities is respectively 1 and 5: The first HR iteration computes 288 qualitative relations in both cases. Differently, during the other iterations 48 qualifications are computed at maximum in the former case, while the latter case requires up to 285 qualifications. The growth is caused by the fact that relations between precise objects are qualified only once: hence, the number of relations to qualify increases with the number of regions imprecisely described.

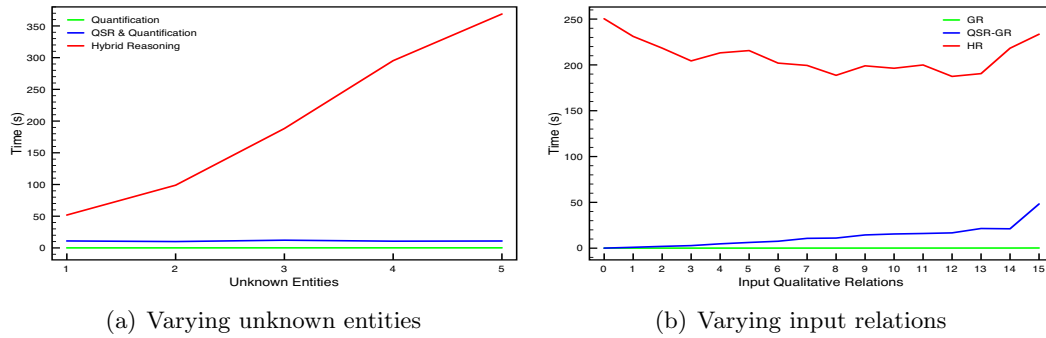


Figure 7.18: Hybrid reasoning system – Computation time.

Finally, Fig. 7.18 shows that HR requires up to 37 times the time necessary for the computation of only QSR-GR. The computation time overload is acceptable for background computation operations only if the hybrid reasoner yields better results than the other components, as the next sections will show.

7.2.3.3 Inferred Qualitative Relations

The experiment outcomes with respect to the number of inferred qualitative spatial relations are summarized in Table 7.1 (dependency on the number of unknown entities) and in Table 7.2 (dependency on the number of input relations). Note that GR does not yield results on the qualitative level, thus it is not reported in the table.

Table 7.1: Output relations – Variable number of unknown entities.

Unknown Entities	QSR-GR		HR	
	Base Relations	Disjunctive Relations	Base Relations	Disjunctive Relations
1	10	3	211	14
2	11	2	159	14
3	11	2	119	12
4	11	2	94	8
5	11	2	83	4

Table 7.2: Output relations – Variable number of input relations.

Input Relations	QSR-GR		HR	
	Base	Disjunctive	Base	Disjunctive
0	0	0	106	0
1	1	0	110	1
2	3	0	114	3
3	4	0	119	4
4	5	0	122	6
5	7	0	125	7
6	8	1	128	8
7	10	2	131	11
8	11	2	136	11
9	13	2	138	11
10	14	2	140	13
11	15	3	141	13
12	17	4	145	16
13	18	4	147	16
14	20	5	148	16
15	23	5	155	14

The number of relations inferred by QSR-GR only depends on the number of input relations, due to the fact that the qualitative spatial reasoner component is able to infer more relations if the input set is bigger. Conversely, HR shows a high dependency both on the number of unknown entities and on the number of input relations. The number of inferred qualitative spatial relations is inversely proportional to the number of unknown entities. Indeed, by keeping the total number of objects managed by the system fixed while increasing the number of unknown entities, the qualification components yields a bigger amount of disjunctive (uncertain) relations; thus the number of base (certain) relations it is able to compute decreases. In turn, the uncertainty propagates to the qualitative reasoning stage, and the overall amount of refined relations decreases. Conversely, the set of refined qualitative relations increases proportionally with the number of input relations: in this case, the qualitative reasoning component refines beforehand the qualitative knowledge, hence the geometric reasoning can better approximate the extensions for all unknown entities and, in turn, the qualification produces more refined disjunctive relations.

Finally, in the performed tests QSR-GR yields, on average, 9% of the relations inferred by HR; thus, HR performs better than QSR-GR in terms of qualitative output information.

7.2.3.4 Quantification of Unknown Entities

The last measure used to evaluate the outcomes is the quantification of unknown entities. The number of tests for which respectively QSR-GR and HR produced better approximations for unknown entities with respect to GR directly applied to the input relations has been measured. In the cases in which they produced better approximations, the percentage of area reduction has been recorded. Since the spatial-region objects can have infinite extent, a workspace with an area 300 times bigger than the area of the MBR that bounds all objects in the HSR input sets has been set in order to limit the objects maximal extension, and evaluate the system in terms of quantification areas.

Table 7.3: Quantification reduction with variable number of unknown entities.

Unknown Entities	Quantifications Reduced (%)		Quantification Reduction (%)	
	QSR-GR	HR	QSR-GR	HR
1	68	84	53	43
2	78	96	43	35
3	74	88	31	27
4	68	80	21	18
5	53	55	14	14

Table 7.4: Quantification reduction with variable number of input relations.

Input Relations	Quantifications Reduced (%)		Quantification Reduction (%)	
	QSR-GR	HR	QSR-GR	HR
1	13	73	1	1
2	33	67	27	14
3	38	66	31	18
4	45	68	29	19
5	59	77	26	20
6	66	73	30	27
7	73	80	30	28
8	76	84	34	31
9	73	84	35	30
10	81	89	37	34
11	84	88	33	32
12	84	92	43	39
13	90	90	39	39
14	80	87	48	45
15	100	100	45	45

Table 7.3 and Table 7.4 summarize the quantitative outcomes by considering respectively a variable number of unknown entities and a variable number of input relations. *Quantifications Reduced* refers to the percentage of performed tests in which the quantifications of the unknown entities are better approximated by respectively QSR-GR and HR with respect to GR, while *Quantification Reduction* reports the average percentage of area reduction.

For both QSR-GR and HR the number of cases in which better approximation can be produced with respect to GR slightly decreases with the number of unknown entities, while it increases with the number of input relations. This result strongly depends on the number of qualitative output relations described above. Indeed, the more refined relations can be inferred by the system, the more it is able to better approximate the extensions of unknown entities. Furthermore, HR produces better approximations in more cases compared with QSR-GR. This result demonstrates that the hybrid reasoning approach performs better than the single approaches executed separately in terms of quantitative output.

However, the percentage of area reduction decreases for HR with respect to QSR-GR. That means, in those cases in which HR yields better approximations than QSR-GR, the percentage of area reduction is lower than in the other cases.

7.2.3.5 Hybrid Reasoning Iterations

The last analysis has been done on the average number of iterations of HR before the algorithm terminated. Fig. 7.19 reports the results grouped by number of unknown entities and number of input relations. Both graphs show that the number of iterations does not depend on these parameters.

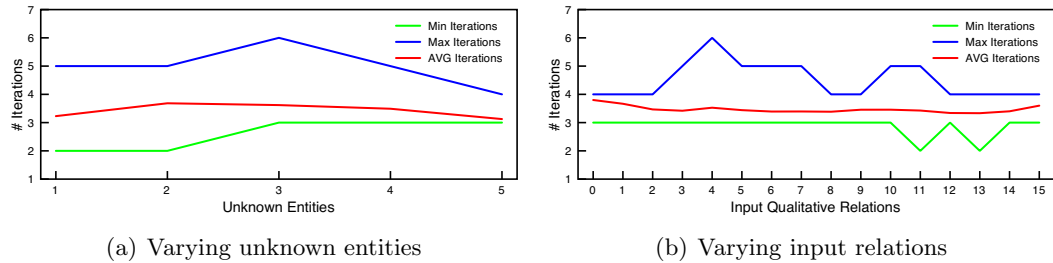


Figure 7.19: Hybrid reasoning system – Reasoning iterations.

Nevertheless, the fact that the number of reasoning iterations is almost constant between three and four further demonstrates that interlinking the two reasoning components (qualitative and geometric reasoning) yields better results than using the approaches individually; otherwise the computation would stop after two iterations.

7.2.3.6 Discussion

The experiments described above aimed at evaluating the hybrid reasoning system capabilities adopting three different measures: computation time, qualitative output, and quantitative output. In particular, the target of the experiments was to empirically prove the claim that the hybrid system yields more precise output with respect to the

single reasoning approaches executed separately, both at the qualitative level (refined qualitative spatial relations) and at the quantitative level (better approximation of unknown entities extension). The empirical results demonstrate the claim. Indeed, at the qualitative level the number of refined qualitative relations that HR yields is up to 10 times bigger than the relations computed by QSR-GR. Furthermore, HR is able to produce better approximations than QSR-GR for the extensions of unknown entities, even if the difference is not always significant. As drawback, in the actual implementation QSR-GR performs many orders of magnitude better than HR in terms of required computation time. The computation time required by HR—background computing operations—grows very rapidly with the number of total entities managed by the system.

7.3 Summary

In this chapter, a prototypical implementation of the geographic information integration system developed in the previous chapters has been shown. In particular, the actual implementation of the system is integrated onto an existing open source GIS. It adds instruments for managing qualitative spatial information as well as for reasoning in a mixed spatial information setting (the spatial information is given partly quantitatively and partly qualitatively).

The prototypical implementation has been adopted for empirically testing the characteristics of the algorithms developed in Chapters 4-6. The computation time requirements for quantifying a single qualitative relation has been shown to be acceptable for single-request operations. The same has been shown for time requirements of the qualification procedures, that compute the qualitative spatial relation between the elements of a single spatial-region objects tuple.

Moreover, it has been shown that the developed hybrid reasoning system performs better compared to the inference techniques developed separately for dealing with either qualitative or quantitative information. The output benefits have been shown both at the qualitative and at the quantitative levels.

Chapter 8

Summary and Outlook

This chapter summarizes and discusses the results achieved in this thesis. An overview of future research based on the described results is given.

8.1 Summary of the Results

Actions taken for responding to natural or human-driven extreme events ask for the support of up-to-date spatial information. However, spatial information collected before the event can no longer be considered reliable: indeed, the extreme event itself causes transformations in the spatial environment. Traditional methods for geospatial data collection are not able to fulfill the information demand-provision gap that follows extreme events rapidly enough (MacFarlane, 2005). New data collection methods need to be developed instead. In recent years, social networks as well as VGI applications have been proposed as alternative for gathering spatial information after extreme events. However, the suggested approaches are strictly linked to traditional collection methods (e.g., GPS). Thus, the information provision gap fulfillment still represents a challenge.

As alternative source of spatial information, communications interchanged among all actors involved in the response operations can be taken into account. Indeed, verbal reports often convey spatial descriptions of the environment struck by the extreme event. Such communications can be potentially interpreted to extract spatial descriptions, that can be used for updating existing information (stored in Geographic Information Systems). However, spatial information conveyed by human reports (verbal or written) has *qualitative* characteristics, that strongly differ from the *quantitative* nature of spatial information stored in GIS. Hence, methodologies for integrating qualitative and quantitative spatial information are required in order to exploit human

communications for updating existing spatial knowledge. In particular, two main targets have been detected: identification of quantitative description for entities that are only qualitatively described, and retrieval of qualitative information that relates entities quantitatively described. The former allows for overlaying on a map the information interpreted from verbal communications, while the latter can trigger warning messages to people involved in decision making operations.

A system for the integration of qualitative and quantitative spatial information has been proposed in this work. Two layers constitute the system: The *Storage Layer* stores spatial information composed by a set of known entities, whose precise quantitative descriptions are known, a set of unknown entities, whose quantitative descriptions are not known, and a set of qualitative spatial relations that relates entities belonging to the two previous sets. The *Geographic Information Integration Layer*, instead, performs all operation necessary for the integration of qualitative and quantitative information. The system produces imprecise descriptions for the spatial extent of unknown entities as output, as well as qualitative descriptions that relate the full set of spatial entities.

The focus of this work was on the development of the geographic information integration layer. Three main functionalities that the layer has to provide have been discussed: translation of qualitative information into quantitative descriptions, translation of quantitative information into qualitative relations, and performing inference operations with information given partly qualitatively and partly quantitatively for boosting the spatial knowledge the system is able to produce. All tasks have to take into account the imprecision of spatial descriptions.

8.1.1 The Quantification Operation

The *quantification* process allows for transforming qualitative information into quantitative descriptions. Given a set of spatial entities and a set of qualitative information that relates them, the quantification yields geometric descriptions for all those entities whose precise spatial extent is not known. This process allows for visualizing entities only qualitatively described, for instance overlaying the interpretation over a map (e.g., Fig 8.1). The transformation yields description of entities whose specification is not as precise as if it would be measured in the reality.

The existing literature related to quantification focuses on finding an exemplary quantitative description for a spatial scene described in qualitative terms. However, this approach does not fit with the aim of this work: the target is instead to provide emergency responders and decision makers with descriptions that bound the area where unknown entities can be located. Hence, the quantification of qualitative relations aims at providing minimal and maximal bounds for the extension of unknown entities.

At first, the quantification of a single relation has been analyzed in Chapter 4, where three different qualitative aspects of space have been considered: topological, directional



Figure 8.1: Overlaying of quantifications over existing maps. The images show a simulation performed to illustrate the benefits of the quantification process in an application of path planning after extreme events. Yellow regions represent the maximal extension of *Hazardous Areas*, while red regions represent their minimal extension. The imprecise description of the Hazardous Areas has been overlaid over 2D and 3D Google Earth¹ maps of Downtown Buffalo (New York). Yellow lines describe the paths planned without information about the Hazardous Areas; blue lines are instead the paths chosen in the case imprecise descriptions of the Hazardous Areas were given. Even if the descriptions are imprecise, quantified information about the Hazardous Areas leads to plan safer paths.

and visibility properties. Given a qualitative relation that relates an unknown entity with a set of entities which spatial extent is known (even if such knowledge can be imprecise), the operation always yields an imprecise description for the maximal and minimal extension of the unknown entity, as constrained by the semantics of the specific relation at hand. Later, in Chapter 6, the quantification of a spatial scene qualitatively described has been discussed, that opportunely combines the results returned by the quantification of single relations.

8.1.2 The Qualification Operation

The operation to translate quantitative information into qualitative information gets the name of *qualification*. The qualification of relations between entities precisely described only requires to test the quantitative descriptions against the constraints defined for the relations in a certain qualitative calculus. The same approach can not be adopted if spatial entities are imprecisely described. One approach traditionally used in the literature consists on defining new sets of relations that can hold between imprecise entities; those relations usually takes the name of *fuzzy* relations. However, the entities considered in this work are not fuzzy or vague, rather their description is imprecise.

¹Google Earth: <http://earth.google.com/>

Hence, it is necessary to compute the set containing all *crisp* qualitative relations that can hold between two entities imprecisely described. Such set of relation gets the name of *crisp relation between imprecise objects*.

In Chapter 5, different approaches for qualifying relations between imprecisely described entities have been discussed. Topological, cardinal direction, and visibility relations have been considered. At first, the topological properties of the method used for representing unknown entities has been exploited to compute the topological crisp relation, adopting standard qualitative spatial reasoning algorithms. For other qualitative aspects (visibility and cardinal direction) the properties of the quantification have been exploited to develop a general approach for qualifying projective relations that can be directly adjusted to other qualitative calculi. However, this approach is not always efficient in practical terms, as has been shown for the qualification of cardinal direction relations. A different theory for qualification has instead been developed for this case, that exploits the structure of the cardinal direction calculus.

8.1.3 Reasoning with Mixed Representations of Spatial Knowledge

In the literature, reasoning techniques exist to perform inference operations with qualitative spatial information. Similarly, computational geometry algorithms have been developed to deal with quantitative spatial information. As these techniques strengthen the capabilities of the respective representation approaches, the development of inference techniques able to perform inference operations with mixed representations of spatial knowledge allows for refining the spatial knowledge in a mixed setting.

Thus, a hybrid reasoning system (HR) has been discussed in Chapter 6 that opportunely connects quantification and qualification with qualitative spatial reasoning and computational geometry algorithms. Empirical evaluations described in Chapter 7 demonstrate that the hybrid reasoning system actually yields more specific information (both on the qualitative and the quantitative side) than the single approaches (QSR-GR) are able to produce individually.

8.2 Outlook

An outlook on promising research directions that follow the approaches proposed in this text is given in the remainder of this chapter.

QUANTIFICATION.

The definition of constructive procedures to compute quantifications of qualitative relations so far had to be conducted on a case-by-case analysis. In general, to be able to quickly integrate new qualitative spatial calculi in the system, an automatic

approach to perform this analysis to construct quantification in the case spatial entities are imprecisely described is desirable.

HYBRID REASONING SYSTEM.

The hybrid reasoning system developed in this work has been shown to produce, in general, more specific information than the qualitative and geometric reasoning components individually. However, HR's requirements in terms of time are orders of magnitude bigger than in the other approaches. Furthermore, there exist cases where HR yields the same results (in terms of quantitative output) as QSR-GR. The overall performance could be improved if an analysis is done in order to identify which characteristics of the input set drive more specific output from HR. Thus, HR can be run only when certain conditions are met by the input set.

THEMATIC INFORMATION EXPLOITATION.

Chapter 6 has been concluded by showing the *thematic reduction* component that allows for reducing the amount of qualitative information pieces before the hybrid reasoning system performs inference operations in mixed setting. The design of the component can be enforced by defining qualitative rules through a formal language. Formal languages allows the system for using standard conceptual reasoning procedures to automatically propagate the rules.

INCONSISTENT SPATIAL INFORMATION.

This work assumes that the system's input information is consistent. This means, there are no pieces of information that contradict each other. However, this is not always the case in real applications. For instance, the communications among rescuers after an extreme event could convey contradicting information due to either wrong evaluations of the entities' properties or to misinterpretation of the received communications. Thus, the system would gain if techniques for checking and eventually resolving inconsistency would be applied to the input information set.

THE TEMPORAL DOMAIN.

In this work, it has been assumed that the spatial entities described in the communications between the responders have a *static* nature. That means, they do not change their spatial extent with time. However this is a strong assumption if entities such as *fire* are considered, that show *dynamic* evolution. Hence, a challenging research direction is the extension of the developed approaches by considering also the temporal domain.

EMERGENCY MANAGEMENT.

The proposed system so far has been evaluated considering geographic information describing an environment non-affected by extreme events. Even though this provides already good measures of the system capabilities, the development of a real application for supporting emergency management passes through the analysis of information available after real events. This knowledge would allow for focusing the system development only on those spatial aspects actually required by responders and decision makers after extreme events.

Appendix A

Infinite-Region Objects: Representation and Algorithms

In Chapter 3 and Chapter 4, the necessity to describe and manage spatial regions with infinite extent arose as a requirement for the development of the quantification component. This appendix discusses the properties of the *infinite-region objects*, and sketches the algorithms for computing intersection, union, and difference of infinite-region objects as well as between a finite polygon and an infinite-region object. Furthermore, the definitions of the minimum bounding rectangle, convex hull, and mutual tangents between infinite-region objects will be discussed.

A.1 Infinite-Region Objects

In Section 4.3 the *infinite-region objects* have been defined as a way to describe regions having infinite extent. The representation is equivalent to the half-plane representation (Rigaux & Scholl, 1995; Rigaux *et al.*, 2002), even though it does not expressly store the parameters of the half-plane equations as proposed by Frank *et al.* (1996); Haunold *et al.* (1997).

A *simple infinite-region object* IR has been defined as a triple $(\Lambda, \vec{\lambda}_1, \vec{\lambda}_2)$, in which the polyline $\Lambda = \langle p_1, \dots, p_n \rangle$ represents the finite boundary of IR , while the two rays $\vec{\lambda}_1 = [p_1, q)$ and $\vec{\lambda}_2 = [p_n, r)$ define the boundaries of IR that extend to infinity. The starting point of Λ corresponds with the starting point of $\vec{\lambda}_1$. In the same way, the last point in Λ corresponds to the starting point of $\vec{\lambda}_2$. An infinite-region object introduces a partition of the space into two infinite regions. The actually represented region is the intersection of the half-plane right of $\vec{\lambda}_1$, the half-plane left of $\vec{\lambda}_2$ and what can intuitively be seen as the area left of the polyline Λ .

The infinite-region representation approach also is suitable to represent polygons with finite extent (Fig. A.1(a)): if the rays $\vec{\lambda}_1$ and $\vec{\lambda}_2$ are not divergent or parallel, they intersect with each other in the point p_{INT} . Thus, an infinite-region object whose rays intersect in p_{INT} is equivalent to the polygon—represented with the vector approach— $P = \langle p_1, \dots, p_n, p_{INT}, p_1 \rangle$. Moreover, two special representation are introduced to model the spatial regions corresponding to \mathbb{R}^2 and \emptyset (Fig. A.1(b)). It is assumed that \mathbb{R}^2 is represented as an infinite-region object, having an empty finite boundary and empty rays. \emptyset is represented as an infinite-region object having an arbitrary point p as its finite boundary, and for which both rays irradiate in an arbitrary and coincident direction from p .

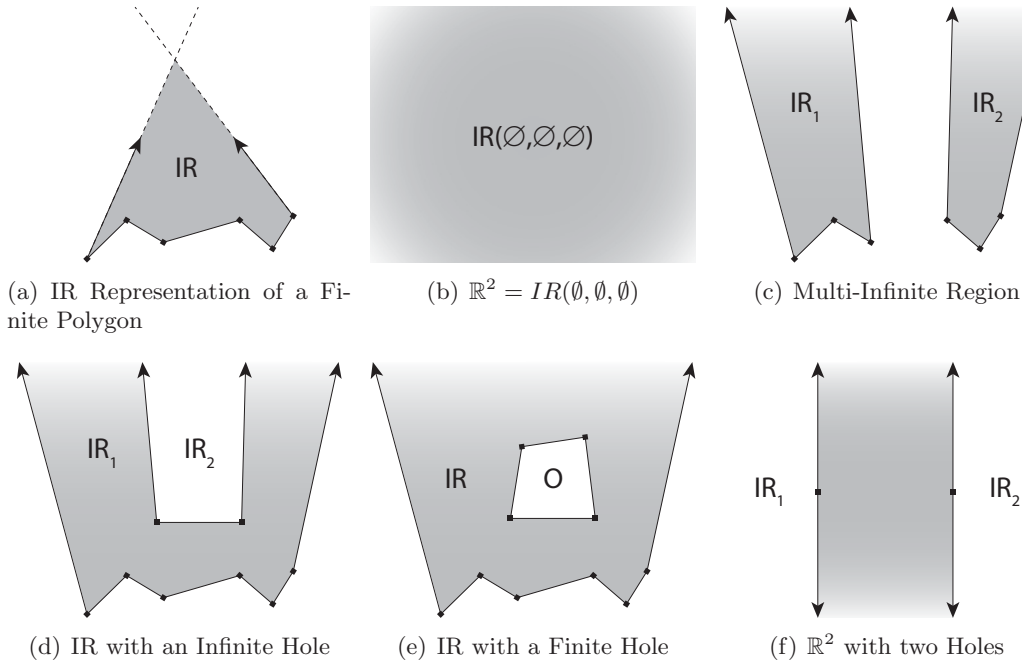


Figure A.1: Complex infinite-region objects: polygons, multi-regions, and regions with holes.

As for the multi-polygon representation, disconnected components of an infinite-region object can be represented by a list of infinite-region objects, e.g., $MIR = \langle IR_1, \dots, IR_n \rangle$ represents an infinite region for which the infinite-region components are described by IR_1, \dots, IR_n . Similarly, infinite-region objects admit holes, that in turn can be infinite or finite. Some examples are depicted in Fig. A.1. In particular, Fig. A.1(c) shows a multi-infinite-region object; Fig. A.1(d) and Fig. A.1(e) depict two infinite-region objects with holes (respectively infinite and finite); finally, the shaded area depicted in Fig. A.1(f), resulting for instance from the quantification of the rela-

tion $N:B:S(O_1, O_2)$, can be represented using the infinite region formalism as \mathbb{R}^2 with two infinite holes corresponding to IR_1 and IR_2 ; hence, $IR = \langle \mathbb{R}^2, IR_1, IR_2 \rangle$.

A.2 Algorithms for Infinite-Region Objects

In this section, computational geometry algorithms are analyzed for computing intersection, union, and difference of infinite-region objects. These operation, however, make use of an auxiliary function for cropping an infinite-region object, hence this operation will be introduced first. Finally, the possibility to define the MBR, convex-hull, and mutual tangents between infinite-region objects will be discussed.

A.2.1 Infinite-Region Object Cropping

Computations necessary to determine the intersection, union, or difference of two infinite-region objects rely on the definition of the *cropping* operation, that is a function to compute a polygon that represents the intersection of an infinite-region object with a given clipping box.

Given an infinite-region object IR , the intersection of IR and a rectangular clipping box CB can be trivially computed by firstly identifying the points i_1 and i_2 where respectively $\vec{\lambda}_1$ and $\vec{\lambda}_2$ crossed the boundary of CB ; the intersection is a finite polygon, which boundary is defined by Λ plus the intersection points of the rays $\vec{\lambda}_1$ and $\vec{\lambda}_2$ with the boundary of CB . An example is depicted in Fig. A.2. The computational time complexity of the cropping operation is $\mathcal{O}(1)$, since it only requires to identify the intersection points of a constant number of rays with a constant number of polygon edges.

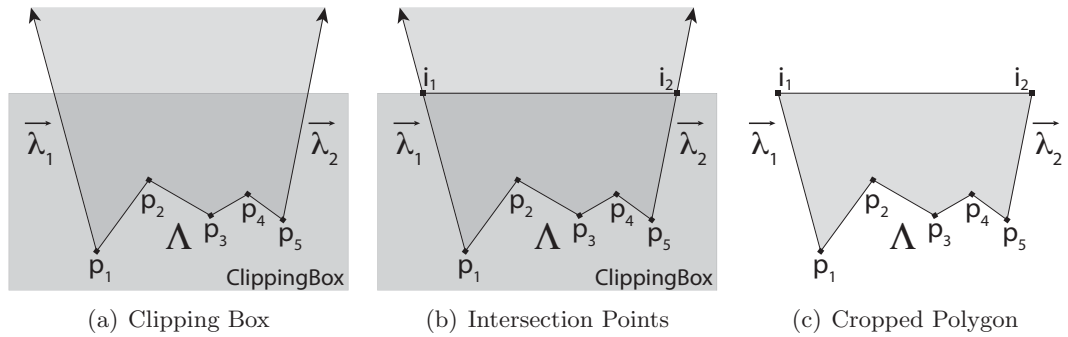


Figure A.2: Cropping of an infinite-region object.

A.2.2 Infinite-Region Objects Intersection

The intersection of two infinite-region objects is realized by first transforming the two objects into simple polygons, performing then the standard intersection of polygons, and finally, if necessary, transforming the result back to an infinite-region object. The intersection yields either a new infinite-region object or a finite polygon defined by one or more components. Let $IR_1 = (\Lambda_{IR_1}, \vec{\lambda}_{IR_1,1}, \vec{\lambda}_{IR_1,2})$ and $IR_2 = (\Lambda_{IR_2}, \vec{\lambda}_{IR_2,1}, \vec{\lambda}_{IR_2,2})$ be two infinite-region objects, and let α be an arbitrary positive and finite value, the algorithm that computes the intersection of IR_1 and IR_2 is presented in Algorithm 15.

The algorithm firstly transforms the infinite-region objects into finite polygons using the cropping operation defined above; it then computes the intersection of the two resulting polygons. Afterwards, it has to be checked whether the actual intersection of the two infinite-region objects has a finite or an infinite extent. In the first case, any point that defines the object belongs either to Λ_{IR_1} , or to Λ_{IR_2} , or is an intersection point between the rays of the two objects. Hence, the inflation of the *MBR* that bounds the union of Λ_{IR_1} and Λ_{IR_2} with the intersection points of the rays necessarily contains the resulting object. Conversely, in the other case, the object resulting from the standard polygon intersection is tangential to the clipping box. Thus, Algorithm 15 uses the function *Tangential* that checks whether at least one edge of O lays upon one of the edges of CB , meaning that the polygon is tangential to the clipping box. Finally, the function *InfiniteRegion*(O, CB) is used to build the infinite-region object from the polygon resulting from the intersection. The result is an object composed by one or more disconnected components. Even though for the sake of simplicity it has not been specified in the algorithm, if two rays are parallel (or coincident), they have no (respectively infinite) intersection point: in such a case an arbitrary point in the plane can be used as a substitute for their mutual intersection point without affecting the intersection result.

Algorithm 15 *IRIntersection*(IR_1, IR_2)

```

1:  $i_1 \leftarrow \vec{\lambda}_{IR_1,1} \cap \vec{\lambda}_{IR_2,1}, i_2 \leftarrow \vec{\lambda}_{IR_1,1} \cap \vec{\lambda}_{IR_2,2}, i_3 \leftarrow \vec{\lambda}_{IR_1,2} \cap \vec{\lambda}_{IR_2,1}, i_4 \leftarrow \vec{\lambda}_{IR_1,2} \cap \vec{\lambda}_{IR_2,2}$ 

2:  $CB \leftarrow BUF\left(MBR(\langle i_1, i_2, i_3, i_4, \Lambda_{IR_1}, \Lambda_{IR_2} \rangle), \alpha\right)$ 

3:  $O \leftarrow IRCropping(IR_1, CB) \cap IRCropping(IR_2, CB)$ 

4: if Tangential( $O, CB$ ) then
5:   return  $O$ 
6: else
7:   return InfiniteRegion( $O, CB$ )
8: end if
```

An example of the intersection procedure is depicted in Fig. A.3. Given the two infinite-region objects IR_1 and IR_2 shown in Fig. A.3(a), the transformation into simple polygons is done by cropping them with a clipping box big enough to contain all the defining points of the objects themselves as well as all intersection points between the rays marking the regions' boundaries. The intersection points between the rays of IR_1 and IR_2 are computed and the bounding box MBR is derived. Lastly, MBR is enlarged by an arbitrary value α resulting in the clipping box CB , as Fig. A.3(a) shows. Now, IR_1 and IR_2 are cropped with CB resulting in the simple polygons P_1 and P_2 (see Fig. A.3(b)). The intersection of P_1 and P_2 is then computed. The result will be a multi-region object which might have more than one component. In the example only a single component is obtained. For each component, it has to be checked whether its polygon needs to be transformed back into an infinite-region object. The final result is shown in Fig. A.3(c). If one of the original objects IR_1 and IR_2 already is a simple polygon, the intersection algorithm works in the same way except that no cropping is required for this object.

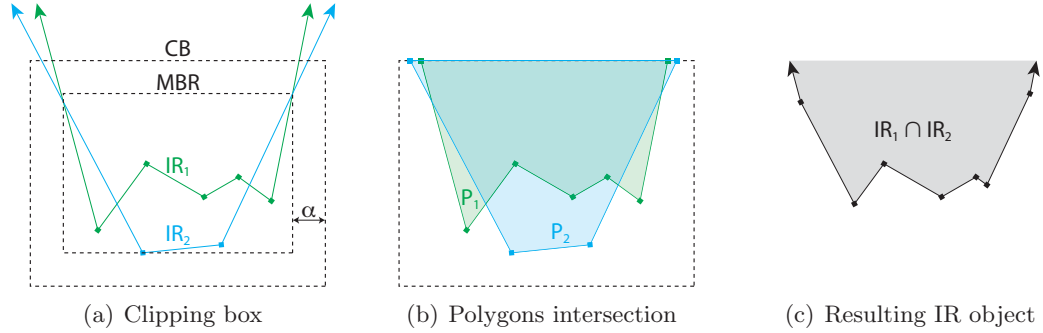


Figure A.3: Intersection of two infinite-region objects.

Computational Complexity of Infinite-Region Objects Intersection

The computational complexity is evaluated considering as input for Algorithm 15 two infinite-region objects IR_1 and IR_2 , represented respectively by n and m points. Since two points define the end-points for the rays, $n-2$ points describe Λ_1 and $m-2$ describe Λ_2 . The steps performed by the algorithm to compute the intersection are:

1. Intersection of infinite-region objects' rays (Line 1). The intersection is computed through algebraic line intersection equations. Hence, this step runs in $\mathcal{O}(1)$ time.
2. The MBR of the union of the intersection points got in Step 1 with the finite boundaries of the two infinite-region objects is computed (Line 2). As shown in Section 2.1.2.1, this operation requires $\mathcal{O}(n + m)$ time.

3. *MBR*, computed in Step 2, is enlarged by a value α (Line 2). The buffering operation has a quadratic time complexity (Section 2.1.2.3). However, *MBR* has a constant number of vertices, hence this step requires $\mathcal{O}(1)$ time.
4. Cropping IR_1 and IR_2 (Line 3). This step runs in $\mathcal{O}(1)$ time.
5. Standard polygon intersection between the results of the cropping operations (Line 3). Since the number of vertices of the cropped polygon is the same as the infinite-region object, the time complexity of this operation is $\mathcal{O}(nm)$. Moreover, the resulting polygon is described by $2(n + m)$ vertices in the worst case.
6. Test if the result of Step 5 is tangential to the clipping box (Line 4). The function *Tangential*(O, CB) has to test whether at least one edge of O exists that lies on one of the edges of CB . Since CB has a constant number of edges, the computational time is $\mathcal{O}(n + m)$.
7. Computation of an infinite-region object (Line 7). The *InfiniteRegion* creator function iterates over all edges of O in order to remove the ones that lie on the boundary of CB . Another iteration is required to build the infinite-region object representation for every resulting component. The computational time complexity of this step is hence $\mathcal{O}(n + m)$.

Overall, the computational time complexity of Algorithm 15 is, in the worst case, $\mathcal{O}(nm)$, that is the same complexity as for the computation of the intersection of two finite polygons (Margalit & Knott, 1989).

A.2.3 Infinite-Region Objects Union and Difference

The algorithms to perform the union and the difference of two infinite-region objects is based on the same principle described for the intersection operation that has been discussed above. However, differently from the intersection case, the union operation never yields a finite polygon. The procedure to compute the union of infinite-region objects is described in Algorithm 16. The difference algorithm, instead, is obtained from Algorithm 15 by replacing the operation \cap with \setminus in Line 3. As for the *IRIntersection* algorithm, the time complexity of the infinite-region objects intersection and difference algorithms is $\mathcal{O}(nm)$ in the worst case.

Algorithm 16 *IRUnion*(IR_1, IR_2)

```

 $i_1 \leftarrow \vec{\lambda}_{IR_1,1} \cap \vec{\lambda}_{IR_2,1}, i_2 \leftarrow \vec{\lambda}_{IR_1,1} \cap \vec{\lambda}_{IR_2,2}, i_3 \leftarrow \vec{\lambda}_{IR_1,2} \cap \vec{\lambda}_{IR_2,1}, i_4 \leftarrow \vec{\lambda}_{IR_1,2} \cap \vec{\lambda}_{IR_2,2}$ 
 $CB \leftarrow BUF\left(MBR(\langle i_1, i_2, i_3, i_4, \Lambda_{IR_1}, \Lambda_{IR_2} \rangle), \alpha\right)$ 
 $O \leftarrow IRCropping(IR_1, CB) \cup IRCropping(IR_2, CB)$ 
return InfiniteRegion( $O, CB$ )

```

A.2.4 MBR, Convex Hull, and Tangents of Infinite-Region Objects

The definition of the Minimum Bounding Rectangle (Section 2.1.2.1) does not undergo any change when an infinite-region object is considered. Indeed, the x-axis and y-axis maximum and minimum values can be equal to ∞ or $-\infty$. Considering, for example, the infinite-region object IR depicted in Fig. A.4(a), its x-axis maximum value is $\overline{X}(IR) = +\infty$, while the other maximum and minimum values are real numbers. The computation of the *MBR* requires only to analyze the direction of the infinite-region object's rays and, consequently, set to ∞ or $-\infty$ the corresponding values. The computational time complexity is hence the same as the finite region case.

Conversely, the convex hull can not be defined for all infinite-region objects. Considering for instance the two infinite-region objects in Fig. A.4; the convex hull of the shaded area in Fig A.4(b) can, in principle, be defined as in Section 2.1.2.2 resulting in the infinite-region object having a red boundary in the image. Instead, for the object in Fig. A.4(c) the definition of convex hull for finite polygon can not be propagated. Hence, it is not possible to define the convex hull for infinite-region objects.

Similarly, the common tangents' definition (Section 2.1.2.5) loses its semantics when one of the objects represent a region with infinite extent, since no tangent exists between a finite and an infinite region. Hence, they can not be computed when one of the input objects represents a region with infinite extent.

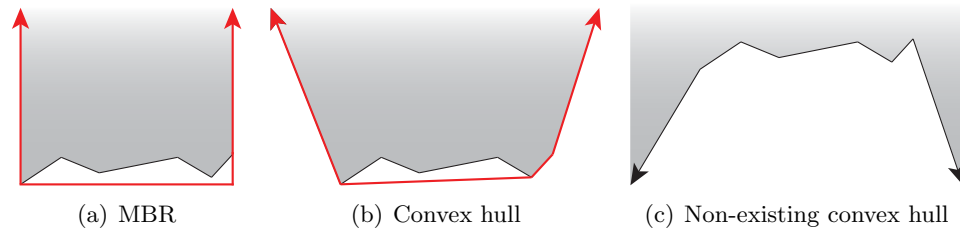


Figure A.4: MBR and convex hull of infinite-region objects.

Appendix B

Testbed – Complementary Information

This appendix provides complementary information about the testbed used for the empirical evaluation, described in Chapter 7, of the qualification component and of the hybrid reasoning system.

Qualification Testbed The qualification testbed is based on 90 geometries selected from the OpenStreetMap¹ dataset of Bremen. All geometries describe the footprint of buildings in the Findorff district in Bremen. The original geometries have been opportunely modified for simulating imprecise descriptions of spatial entities. The details of the modification made over the original geometries are explained in Section 7.2.2.

The qualification testbed (cf. Fig. 7.12), overlaid on a *Google Maps*² map of the Findorff district, is shown in Fig. B.1. The top-left corner of the map has coordinates (Latitude, Longitude): $53^{\circ}05'55''N$, $8^{\circ}48'12''E$. The bottom-right corner of the map has coordinates: $53^{\circ}05'23''N$, $8^{\circ}49'06''E$.

Hybrid Reasoning System Testbed The hybrid reasoning system has been evaluated over a testbed based on six geometries extracted from the OpenStreetMap dataset of Bremen (cf. Section 7.2.3). All geometries describe the footprint of buildings in the Findorff district.

The hybrid reasoning system testbed (cf. Fig. 7.16), overlaid on a *Google Maps* map, is shown in Fig. B.2. The top-left corner of the map has coordinates (Latitude, Longitude): $53^{\circ}05'38''N$, $8^{\circ}48'33''E$. The bottom-right corner of the map has coordinates: $53^{\circ}05'28''N$, $8^{\circ}49'05''E$.

¹OpenStreetMap: <http://www.openstreetmap.org/>

²Google Maps: <http://maps.google.com/>

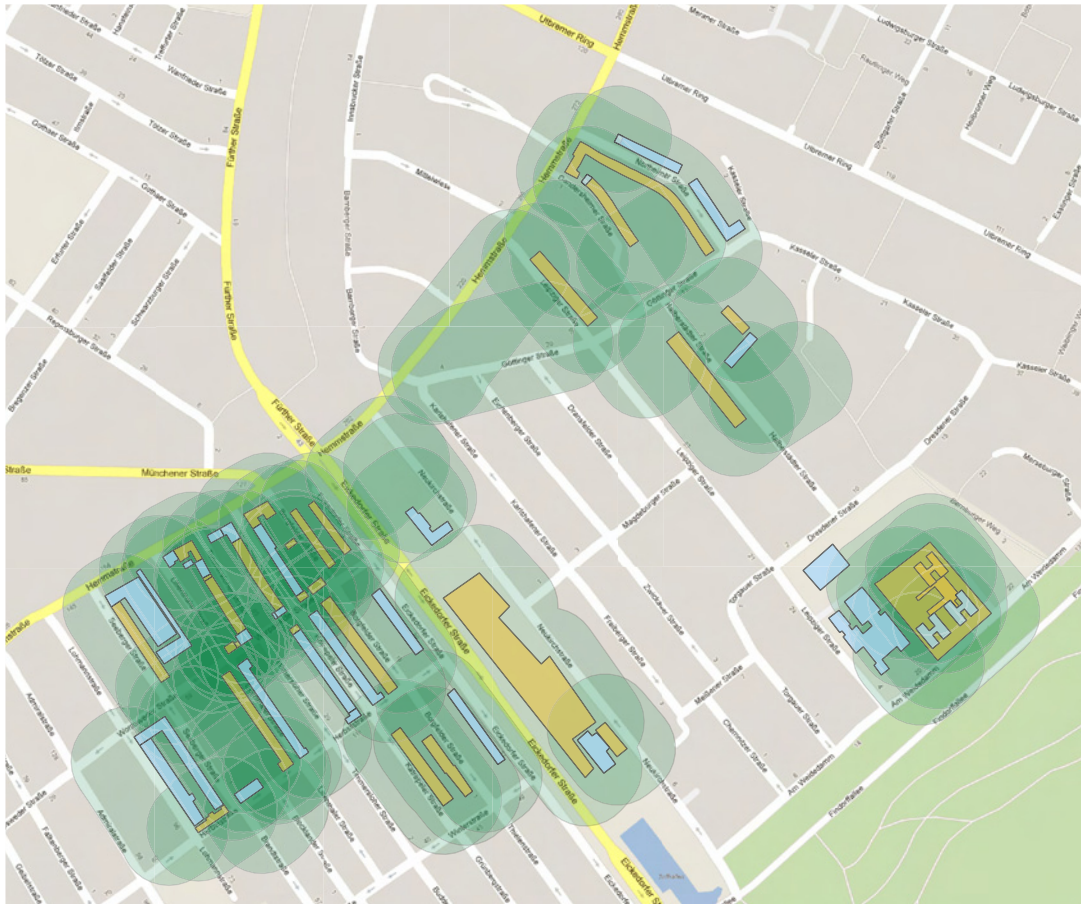


Figure B.1: Qualification testbed.

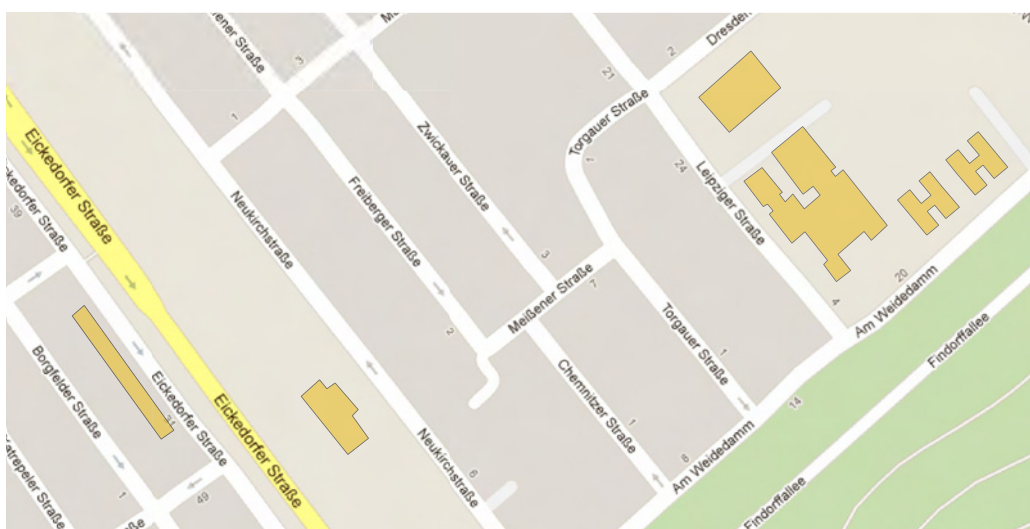


Figure B.2: Hybrid spatial reasoning testbed.

References

- ALLEN, J. (1983). Maintaining knowledge about temporal intervals. *Communications of the ACM*, **26**(11), 832–843. (35)
- ALTMAN, D. (1994). Fuzzy set theoretic approaches for handling imprecision in spatial analysis. *International journal of geographical information systems*, **8**(3), 271–289. (41, 42)
- BALBIANI, P., CONDOTTA, J.F. & FARIÑAS DEL CERRO, L. (1998). A model for reasoning about bidimensional temporal relations. In L.S. A. G. Cohn & S.C. Shapiro, eds., *Proceedings of the Sixth International Conference on Principles of Knowledge Representation and Reasoning*, 124–130, Morgan Kaufmann. (21, 31, 35, 39)
- BEAUBOUEF, T., PETRY, F.E. & LADNER, R. (2007). Spatial data methods and vague regions: A rough set approach. *Applied Soft Computing*, **7**(1), 425–440. (43, 44)
- BERENDT, B., BARKOWSKY, T., FREKSA, C. & KELTER, S. (1998). Spatial representation with aspect maps. In C. Freksa, C. Habel & K. Wender, eds., *Spatial Cognition*, vol. 1404 of *Lecture Notes in Computer Science*, 313–336, Springer Berlin / Heidelberg. (20)
- BILLEN, R. & CLEMENTINI, E. (2004). A model for ternary projective relations between regions. In E. Bertino, S. Christodoulakis, D. Plexousakis, V. Christophides, M. Koubarakis, K. Böhm & E. Ferrari, eds., *Advances in Database Technology - EDBT 2004, Proceedings of the 9th International Conference on Extending Database Technology*, vol. 2992 of *Lecture Notes in Computer Science*, 310–328, Springer. (21, 22, 36, 37, 40)
- BILLEN, R. & CLEMENTINI, E. (2006). Projective relations in a 3D environment. In M. Raubal, H.J. Miller, A.U. Frank & M.F. Goodchild, eds., *Proceedings of the 4th International Conference on Geographic Information Science, GIScience 2006*, vol. 4197 of *Lecture Notes in Computer Science*, 18–32, Springer. (21)
- BILLEN, R. & KURATA, Y. (2008). Refining topological relations between regions considering their shapes. In T. Cova, H. Miller, K. Beard, A. Frank & M. Goodchild, eds., *Proceedings of the 5th International Conference on Geographic Information Science*, vol. 5266 of *Lecture Notes in Computer Science*, 20–37, Springer Berlin / Heidelberg. (39, 145)
- BILLEN, R., ZLATANOVA, S., MATHONET, P. & BONIVER, F. (2002). The dimensional model: a framework to distinguish spatial relationships. In D. Richardson & P. van Oosterom, eds., *Advances in Spatial Data handling*, 285–298, Springer-Verlag. (21, 39)

- BIPM, IEC, IFCC, ILAC, ISO, IUPAC, IUPAP & OIML (2008). *International vocabulary of metrology - Basic and general concepts and associated terms (VIM)*. ISO, Geneva. (41)
- BITTNER, T. & SMITH, B. (2003). Vague reference and approximating judgments. *Spatial Cognition and Computation*, **3**(2), 137–156. (42)
- BITTNER, T. & STELL, J. (2001). Rough sets in approximate spatial reasoning. In W. Ziarko & Y. Yao, eds., *Rough Sets and Current Trends in Computing*, vol. 2005 of *Lecture Notes in Computer Science*, 445–453, Springer Berlin / Heidelberg. (43, 44)
- BRAGEUL, D. & GUESGEN, H.W. (2007). A model for qualitative spatial reasoning combining topology, orientation and distance. In D. Wilson & G. Sutcliffe, eds., *FLAIRS Conference*, 653–658, AAAI Press. (39, 145)
- CALVANESE, D., DE GIACOMO, G., LENZERINI, M. & NARDI, D. (2001). Reasoning in expressive description logics. In A. Robinson & A. Voronkov, eds., *Handbook of Automated Reasoning*, 1581–1634, Elsevier and MIT Press. (148)
- CARGILE, J. (1969). The sorites paradox. *The British Journal for the Philosophy of Science*, **20**(3), 193–202. (42)
- CHEN, J., JIA, H., LIU, D. & ZHANG, C. (2010). Inversing cardinal direction relations. In *Proceedings of the 2010 Fifth International Conference on Frontier of Computer Science and Technology*, FCST '10, 276–281, IEEE Computer Society, Washington, DC, USA. (35)
- CICERONE, S. & DI FELICE, P. (2000). Cardinal relations between regions with a broad boundary. In *Proceedings of the 8th ACM International Symposium on Advances in Geographic Information Systems*, GIS '00, 15–20, ACM, New York, NY, USA. (44, 65, 104, 122, 124)
- CICERONE, S. & DI FELICE, P. (2004). Cardinal directions between spatial objects: the pairwise-consistency problem. *Information Sciences*, **164**(1-4), 165–188. (32, 35)
- CLEMENTINI, E. & BILLEN, R. (2006). Modeling and computing ternary projective relations between regions. *IEEE Transactions on Knowledge and Data Engineering*, **18**(6), 799–814. (17, 21, 40, 76)
- CLEMENTINI, E. & DI FELICE, P. (1996). An algebraic model for spatial objects with indeterminate boundaries. In P.A. Burrough & A.U. Frank, eds., *Geographic Objects with Indeterminate Boundaries*, GISDATA Series, 155–169, London: Taylor & Francis. (42, 44, 122)
- CLEMENTINI, E. & DI FELICE, P. (1997a). Approximate topological relations. *International Journal of Approximate Reasoning*, **16**(2), 173–204. (42, 43, 44)
- CLEMENTINI, E. & DI FELICE, P. (1997b). A global framework for qualitative shape description. *GeoInformatica*, **1**(1), 11–27. (21)
- CLEMENTINI, E. & DI FELICE, P. (2001). A spatial model for complex objects with a broad boundary supporting queries on uncertain data. *Data Knowledge and Engineering*, **37**(3), 285–305. (42, 43, 44)

- CLEMENTINI, E., DI FELICE, P. & HERNÁNDEZ, D. (1997). Qualitative representation of positional information. *Artificial Intelligence*, **95**(2), 317–356. (39, 146)
- COHN, A. & GOTTS, N. (1996). The ‘egg-yolk’ representation of regions with indeterminate boundaries. In *Geographical Objects with Undetermined Boundaries*, vol. 2, 171–187, Francis Taylor. (42, 44, 60, 65, 104, 105, 128)
- COHN, A. & VARZI, A. (1999). Modes of connection. In C. Freksa & D. Mark, eds., *Spatial Information Theory. Cognitive and Computational Foundations of Geographic Information Science*, vol. 1661 of *Lecture Notes in Computer Science*, 749–749, Springer Berlin / Heidelberg. (29)
- COHN, A.G. (1997). Qualitative spatial representation and reasoning techniques. In G. Brewka, C. Habel & B. Nebel, eds., *Proceedings of the 21st Annual German Conference on Artificial Intelligence: Advances in Artificial Intelligence*, vol. 1303 of *Lecture Notes in Computer Science*, 1–30, Springer Berlin / Heidelberg. (21)
- COHN, A.G. & HAZARIKA, S.M. (2001). Qualitative spatial representation and reasoning: An overview. *Fundamenta Informaticae*, **46**(1-2), 1–29. (21)
- COHN, A.G. & RENZ, J. (2008). Qualitative spatial representation and reasoning. In V.L. Frank van Harmelen & B. Porter, eds., *Handbook of Knowledge Representation*, vol. 3 of *Foundations of Artificial Intelligence*, 551–596, Elsevier. (21)
- COHN, A.G., BENNETT, B., GOODAY, J. & GOTTS, N. (1997). RCC: a calculus for region-based qualitative spatial reasoning. *GeoInformatica*, **1**, 275–316. (21, 30)
- CONDOTTA, J.F., LIGOZAT, G. & SAADE, M. (2006). A generic toolkit for n-ary qualitative temporal and spatial calculi. In *TIME 2006. Thirteenth International Symposium on Temporal Representation and Reasoning*, 78–86. (25)
- COWEN, D.J. (1988). GIS versus CAD versus DBMS: What are the differences. *Photogrammetric Engineering and Remote Sensing*, **54**(11), 1551–1554. (11)
- CUI, Z., COHN, A.G. & RANDELL, D.A. (1993). Qualitative and topological relationships in spatial databases. In D. Abel & B. Chin Ooi, eds., *Proceedings of the Third International Symposium on Advances in Spatial Databases*, vol. 692 of *Lecture Notes in Computer Science*, 296–315, Springer Berlin / Heidelberg. (30)
- DE BERG, M., CHEONG, O., VAN KREVELD, M. & OVERMARS, M. (2008). *Computational geometry: algorithms and applications*. Springer-Verlag, 3rd edn. (15)
- DE FELICE, G., FOGLIARONI, P. & WALLGRÜN, J.O. (2010). Qualitative reasoning with visibility information for environmental learning. In R. Purves & R. Weibel, eds., *Extended Abstracts Volume, GIScience 2010*. (38, 138)
- DE FELICE, G., FOGLIARONI, P. & WALLGRN, J. (2011). A hybrid geometric-qualitative spatial reasoning system and its application in gis. In M. Egenhofer, N. Giudice, R. Moratz & M. Worboys, eds., *Spatial Information Theory*, vol. 6899 of *Lecture Notes in Computer Science*, 188–209, Springer Berlin / Heidelberg. (129)

- DILO, A., DE BY, R. & STEIN, A. (2007). A system of types and operators for handling vague spatial objects. *International Journal of Geographical Information Science*, **21**(4), 397–426. (42, 44)
- DONINI, F.M., LENZERINI, M., NARDI, D. & SCHAEFER, A. (1996). Reasoning in description logics. In G. Brewka, ed., *Principles of Knowledge Representation*, 191–236, Center for the Study of Language and Information. (148)
- DRABEK, T.E. & HOETMER, G.J. (1991). *Emergency Management: Principles and Practice for Local Government*. Washington, DC: International City Management Association. (46)
- DU, S. & GUO, L. (2010). Modeling and querying approximate direction relations. *ISPRS Journal of Photogrammetry and Remote Sensing*, **65**(4), 328–340. (44, 122, 124)
- DUCKHAM, M., MASON, K., STELL, J. & WORBOYS, M. (2001). A formal approach to imperfection in geographic information. *Computers, Environment and Urban Systems*, **25**(1), 89–103. (41, 42)
- DUCKHAM, M., LINGHAM, J., MASON, K.T. & WORBOYS, M.F. (2006). Qualitative reasoning about consistency in geographic information. *Information Sciences*, **176**(6), 601–627. (146)
- DÜNTSCH, I., WANG, H. & MCCLOSKEY, S. (2001). A relation - algebraic approach to the region connection calculus. *Theoretical Computer Science*, **255**(1-2), 63–83. (30)
- DUTTON, G. (1992). Handling positional uncertainty in spatial databases. In P. Bresnahan, E. Corwin & D. Cowen, eds., *Proceedings of the 5th International Symposium on Spatial Data Handling*, 460–469, IGU Commission of GIS. (44)
- DYLLA, F. & MORATZ, R. (2004). Empirical complexity issues of practical qualitative spatial reasoning about relative position. In *Workshop on Spatial and Temporal Reasoning at ECAI 2004*. (28, 67, 137)
- EGENHOFER, M. (1989). A formal definition of binary topological relationships. In W. Litwin & H.J. Schek, eds., *Foundations of Data Organization and Algorithms*, vol. 367 of *Lecture Notes in Computer Science*, 457–472, Springer Berlin / Heidelberg. (30)
- EGENHOFER, M.J. (1991). Reasoning about binary topological relations. In O. Günther & H.J. Schek, eds., *Proceedings of the Second International Symposium on Advances in Spatial Databases*, vol. 525 of *Lecture Notes in Computer Science*, 143–160, Springer Berlin / Heidelberg. (21, 31, 39)
- EGENHOFER, M.J. & FRANZOSA, R.D. (1991). Point-set topological spatial relations. *International journal of geographical information systems*, **5**(2), 161–174. (21, 29, 30, 40)
- EGENHOFER, M.J., CLEMENTINI, E. & DI FELICE, P. (1994). Topological relations between regions with holes. *International Journal of Geographical Information Systems*, **8**(2), 129–142. (31)
- ERWIG, M. & SCHNEIDER, M. (1997). Vague regions. In M. Scholl & A. Voisard, eds., *Proceedings of the 5th International Symposium on Advances in Spatial Databases, SSD 1997*, vol. 1262 of *Lecture Notes in Computer Science*, 298–320, Springer. (42, 43, 44)

- FELKEL, P. & OBDRZALEK, S. (1998). Straight skeleton implementation. In L.S. Kalos, ed., *Proceedings of the 14th Spring Conference on Computer Graphics*, 210–218, Comenius University, Bratislava. (16, 162)
- FISHER, P. (2000). Sorites paradox and vague geographies. *Fuzzy Sets and Systems*, **113**(1), 7–18. (31, 42)
- FOGLIARONI, P., WALLGRÜN, J.O., CLEMENTINI, E., TARQUINI, F. & WOLTER, D. (2009). A qualitative approach to localization and navigation based on visibility information. In K.S. Hornsby, C. Claramunt, M. Denis & G. Ligozat, eds., *Proceedings of the 9th International Conference on Spatial Information Theory*, vol. 5756 of *Lecture Notes in Computer Science*, 312–329, Springer Berlin / Heidelberg. (21, 36, 37, 85)
- FONTE, C. & LODWICK, W. (2005). Modelling the fuzzy spatial extent of geographical entities. In F.E. Petry, V.B. Robinson & M.A. Cobb, eds., *Fuzzy Modeling with Spatial Information for Geographic Problems*, 121–142, Springer Berlin Heidelberg. (43)
- FRANK, A.U. (1991). Qualitative spatial reasoning with cardinal directions. In H. Kaindl, ed., *Proceedings of the 7th Austrian Conference on Artificial Intelligence, ÖGAI*, vol. 287 of *Informatik-Fachberichte*, 157–167, Springer. (21, 22, 31)
- FRANK, A.U. (1992). Qualitative spatial reasoning about distances and directions in geographic space. *Journal of Visual Languages and Computing*, **3**(4), 343–371. (39, 145)
- FRANK, A.U. (1996). Qualitative spatial reasoning: cardinal directions as an example. *International journal of geographical information systems*, **10**(3), 269–290. (32)
- FRANK, A.U., HAUNOLD, P., KUHN, W. & KUPER, G. (1996). Representation of geometric objects as set of inequalities. In K.H. Hinrichs, ed., *12th European Workshop on Computational Geometry CG'96*, 51–56, Universität Münster. (73, 181)
- FREKSA, C. (1980). Communication about visual patterns by means of fuzzy characterizations. In *XXIIInd International Congress of Psychology, Leipzig, Germany*. (42)
- FREKSA, C. (1982). Linguistic description of human judgments in expert systems and in the 'soft' sciences. In M. Gupta & E. Sanchez, eds., *Approximate Reasoning in Decision Analysis*, 297–305, North-Holland Publishing Company. (42)
- FREKSA, C. (1991). Qualitative spatial reasoning. In Mark, D.M. and Frank, A.U., ed., *Cognitive and linguistic aspects of geographic space*, 361–372. (54)
- FREKSA, C. (1992). Using orientation information for qualitative spatial reasoning. In A.U. Frank, I. Campari & U. Formentini, eds., *Proceedings of the International Conference GIS - From Space to Territory: Theories and Methods of Spatio-Temporal Reasoning in Geographic Space*, 162–178, Springer-Verlag. (21, 22, 25)
- FREKSA, C. (1994). Fuzzy systems in AI: An overview. In R. Kruse, J. Gebhardt & R. Palm, eds., *Fuzzy systems in computer science*, 155–169, Vieweg, Braunschweig/Wiesbaden. (41, 65)

- FREKSA, C. & LÓPEZ DE MÁNTARAS, R. (1982). An adaptive computer system for linguistic categorization of “soft” observations in expert systems and in the social sciences. In *Proceedings of the 2nd World Conference on Mathematics at the Service of Man, Las Palmas*, 288–292. (42)
- FREKSA, C. & ZIMMERMANN, K. (1993). On the utilization of spatial structures for cognitively plausible and efficient reasoning. In *Proceedings of the Workshop on Spatial and Temporal Reasoning, 13th International Joint Conference on Artificial Intelligence*, 61–66. (24)
- FREKSA, C., MORATZ, R. & BARKOWSKY, T. (2000). Schematic maps for robot navigation. In T. Freksa, C. Habel, W. Brauer & K.F. Wender, eds., *Spatial Cognition II*, vol. 1849 of *Lecture Notes in Computer Science*, 100–114, Springer Berlin, Heidelberg. (21)
- GALTON, A. & MEATHREL, R. (1999). Qualitative outline theory. In *Proceedings of the 16th International Joint Conference on Artificial Intelligence - Volume 2*, 1061–1066, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA. (21)
- GANTNER, Z., WESTPHAL, M. & WÖLFL, S. (2008). GQR – A fast reasoner for binary qualitative constraint calculi. In *Proceedings of the AAAI’08 Workshop on Spatial and Temporal Reasoning*. (131)
- GEREVINI, A. & RENZ, J. (2002). Combining topological and size information for spatial reasoning. *Artificial Intelligence*, **137**(1-2), 1–42. (21, 40, 145)
- GOODCHILD, M. (2003). Geospatial data in emergencies. In S. Cutter, D. Richardson & T. Wilbanks, eds., *The Geographical Dimensions of Terrorism*, 99–104, New York: Routledge. (51)
- GOODCHILD, M.F. (1989). Modeling error in objects and fields. In M.F. Goodchild & S. Gopal, eds., *The Accuracy of Spatial Databases*, 107–113, Taylor & Francis. (12)
- GOODCHILD, M.F. & GLENNON, J.A. (2010). Crowdsourcing geographic information for disaster response: a research frontier. *International Journal of Digital Earth*, **3**(3), 231–241. (51)
- GOODCHILD, M.F., FU, P. & RICH, P. (2007). Sharing geographic information: An assessment of the geospatial one-stop. *Annals of the Association of American Geographers*, **97**(2), 250–266. (20)
- GOYAL, R. & EGENHOFER, M. (2000). Consistent queries over cardinal directions across different levels of detail. In *Proceedings of the 11th International Workshop on Database and Expert System Applications*, 876–880, IEEE Computer Society. (31)
- GOYAL, R. & EGENHOFER, M. (in press). Cardinal directions between extended spatial objects. *IEEE Transactions on Knowledge and Data Engineering*. (21, 22, 31, 40, 76)
- GUESGEN, H.W. (1989). Spatial reasoning based on allen’s temporal logic. Tech. rep., International Computer Science Institute. (21, 31, 35, 40)
- GUESGEN, H.W. & ALBRECHT, J. (2000). Imprecise reasoning in geographic information systems. *Fuzzy Sets and Systems*, **113**(1), 121–131. (42)
- GUO, L. & DU, S. (2009). Deriving topological relations between regions from direction relations. *Journal of Visual Language and Computing*, **20**(6), 368–384. (39, 146)

- HAUNOLD, P., GRUMBACH, S., KUPER, G. & LACROIX, Z. (1997). Linear constraints: Geometric objects represented by inequalities. In S. Hirtle & A. Frank, eds., *Spatial Information Theory: A Theoretical Basis for GIS*, vol. 1329 of *Lecture Notes in Computer Science*, 429–440, Springer Berlin / Heidelberg. (73, 181)
- HERNÁNDEZ, D. (1994). *Qualitative Representation of Spatial Knowledge*, vol. 804 of *Lecture Notes in Computer Science*. Springer. (39)
- HERNÁNDEZ, D., CLEMENTINI, E. & DI FELICE, P. (1995). Qualitative distances. In A. Frank & W. Kuhn, eds., *Spatial Information Theory A Theoretical Basis for GIS*, vol. 988 of *Lecture Notes in Computer Science*, 45–57, Springer Berlin / Heidelberg. (21)
- HERRING, J. (2001). The OpenGIS abstract specification, Topic 1: Feature geometry (ISO 19107 Spatial schema), version 5. *OGC Document*, 01–101. (12, 13, 16, 40, 73)
- HOWE, J. (2008). *Crowdsourcing: Why the Power of the Crowd is Driving the Future of Business*. Crown Business, New York. (20, 51)
- HUGENTOBLE, M. (2008). Quantum GIS. In S. Shekhar & H. Xiong, eds., *Encyclopedia of GIS*, 935–939, Springer. (149)
- ILIFFE, J. & LOTT, R. (2008). *Datums and map projections for remote sensing, GIS and surveying*. Whittles Publishing. (12)
- KEEFE, R. (2000). *Theories of Vagueness*. New York: Cambridge University Press. (42)
- KIRKPATRICK, D. & SNOEYINK, J. (1995). Computing common tangents without a separating line. In S. Akl, F. Dehne, J.R. Sack & N. Santoro, eds., *In Proceedings of the 4th Workshop on Algorithms and Data Structures (WADS)*, vol. 955 of *Lecture Notes in Computer Science*, 183–193, Springer Berlin / Heidelberg. (18, 99)
- LEHMANN, F. & COHN, A.G. (1994). The egg/yolk reliability hierarchy: semantic data integration using sorts with prototypes. In *Proceedings of the Third International Conference on Information and Knowledge Management, CIKM '94*, 272–279, ACM. (42, 60)
- LEVINSON, S.C. (1996). Frames of reference and molyneux's question: Crosslinguistic evidence. In P. Bloom, M.A. Peterson, L. Nadel & M.F. Garrett, eds., *Language and Space*, 109–169, MIT Press, Cambridge, MA. (21)
- LI, L. & GOODCHILD, M.F. (2010). The role of social networks in emergency management: A research agenda. *International Journal of Information Systems for Crisis Response and Management (IJISCRAM)*, **2**(4), 48–58. (51)
- LI, R., BHANU, B., RAVISHANKAR, C., KURTH, M. & NI, J. (2007). Uncertain spatial data handling: Modeling, indexing and query. *Computers & Geosciences*, **33**(1), 42–61. (43)
- LI, S. (2006). Combining topological and directional information: First results. In J. Lang, F. Lin & J. Wang, eds., *Proceedings of the First International Conference on Knowledge Science, Engineering and Management, KSEM 2006*, vol. 4092 of *Lecture Notes in Computer Science*, 252–264, Springer. (39)

- LI, S. (2007). Combining topological and directional information for spatial reasoning. In *Proceedings of the 20th International Joint conference on Artificial Intelligence*, IJCAI'07, 435–440, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA. (40, 145)
- LI, S. & COHN, A.G. (2009). Reasoning with topological and directional spatial information. *CoRR*, **abs/0909.0122**. (40)
- LI, Y. & LI, S. (2004). A fuzzy sets theoretic approach to approximate spatial reasoning. *IEEE Transactions on Fuzzy Systems*, **12**(6), 745–754. (44)
- LIGOZAT, G. (1993). Qualitative triangulation for spatial reasoning. In A.U. Frank & I. Campari, eds., *Proceedings of the International Conference on Spatial Information Theory: A Theoretical Basis for GIS, COSIT '93*, vol. 716 of *Lecture Notes in Computer Science*, 54–68, Springer. (21, 22)
- LIGOZAT, G. & RENZ, J. (2004). What is a qualitative calculus? A general framework. In C. Zhang, H.W. Guesgen & W.K. Yeap, eds., *Proceedings of the 8th Pacific Rim International Conference on Artificial Intelligence: Trends in Artificial Intelligence, PRICAI 2004*, vol. 3157 of *Lecture Notes in Computer Science*, 53–64, Springer Berlin / Heidelberg. (22)
- LIU, K. & SHI, W. (2006). Computing the fuzzy topological relations of spatial objects based on induced fuzzy topology. *International Journal of Geographical Information Science*, **20**(8), 857–883. (42, 44)
- LIU, W., LI, S. & RENZ, J. (2009). Combining RCC-8 with qualitative direction calculi: Algorithms and complexity. In *Proceedings of the 21st International Joint Conference on Artificial Intelligence*, 854–859, Morgan Kaufmann Publishers Inc. (40, 145)
- LONGLEY, P., GOODCHILD, M., MAGUIRE, D. & RHIND, D. (2005). *Geographic Information Systems and Science*. John Wiley & Sons, Ltd. (11, 12, 18, 20)
- MACFARLANE, R. (2005). *A Guide to GIS Applications in Integrated Emergency Management*. Emergency Planning College, Cabinet Office. (47, 48, 175)
- MACKWORTH, A. (1977). Consistency in networks of relations. *Artificial Intelligence*, **8**(1), 99–118. (28, 67, 106, 135, 136)
- MARGALIT, A. & KNOTT, G.D. (1989). An algorithm for computing the union, intersection or difference of two polygons. *Computers & Graphics*, **13**(2), 167–183. (17, 74, 98, 186)
- MARK, D.M. & CSILLAG, F. (1989). The nature of boundaries on area-class maps. *Cartographica*, **26**(1), 65–77. (43)
- MELKMAN, A.A. (1987). On-line construction of the convex hull of a simple polyline. *Information Processing Letters*, **25**(1), 11–12. (16, 99)
- MONTANARI, U. (1974). Networks of constraints: Fundamental properties and applications to picture processing. *Information Science*, **7**(2), 95–132. (28, 135, 136)

- MORATZ, R., RENZ, J. & WOLTER, D. (2000). Qualitative spatial reasoning about line segments. In W. Horn, ed., *Proceedings of the 14th European Conference on Artificial Intelligence, ECAI 2000*, 234–238, IOS Press. (21, 22)
- MORATZ, R., DYLLA, F. & FROMMBERGER, L. (2005). A relative orientation algebra with adjustable granularity. In *Proceedings of the Workshop on Agents in Real-Time and Dynamic Environments (IJCAI 05)*. (21, 22)
- NATIONAL GOVERNORS' ASSOCIATION (1978). *1978 Emergency Preparedness Project: Final Report*. Washington, D.C. (46)
- NYERGES, T.L. & JANKOWSKI, P. (2010). *Regional and urban GIS: a decision support approach*. Guilford Press, New York. (12)
- OKOLLOH, O. (2008). Ushahidi, or 'testimony': Web 2.0 tools for crowdsourcing crisis information. *Participatory Learning and Action*, **59**(1), 65–70. (52)
- O'ROURKE, J. (1998). *Computational Geometry in C*. Cambridge University Press. (15)
- PAPADIAS, D., SELLIS, T., THEODORIDIS, Y. & EGENHOFER, M.J. (1995). Topological relations in the world of minimum bounding rectangles: a study with r-trees. In *Proceedings of the 1995 ACM SIGMOD International Conference on Management of Data*, SIGMOD '95, 92–103, ACM. (21)
- PAULY, A. & SCHNEIDER, M. (2004). Vague spatial data types, set operations, and predicates. In G. Gottlob, A.A. Benczúr & J. Demetrovics, eds., *Proceedings of the 8th East European Conference on Advances in Databases and Information Systems, ADBIS 2004*, vol. 3255 of *Lecture Notes in Computer Science*, 379–392, Springer. (42, 43, 44)
- PAULY, A. & SCHNEIDER, M. (2008). Spatial vagueness and imprecision in databases. In R.L. Wainwright & H. Haddad, eds., *Proceedings of the 2008 ACM Symposium on Applied Computing (SAC)*, 875–879, ACM. (41, 42, 44)
- PAWLAK, Z. (1982). Rough sets. *International Journal of Information and Computer Sciences*, **11**(5), 341–356. (43)
- RAIMAN, O. (1991). Order of magnitude reasoning. *Artificial Intelligence*, **51**(1-3), 11–38. (21)
- RAMSEY, P. (2010). *PostGIS 1.5.2 Manual*. (150)
- RANDELL, D.A., CUI, Z. & COHN, A. (1992). A spatial logic based on regions and connection. In *Proceedings of the 3rd International Conference on Principles of Knowledge Representation and Reasoning*, 165–176, Morgan Kaufmann. (21, 29, 94)
- RENSCHLER, C., FRAZIER, A., ARENDT, L., CIMELLARO, G., REINHORN, A. & BRUNEAU, M. (2010). Developing the 'PEOPLES' resilience framework for defining and measuring disaster resilience at the community scale. In *9th US National and 10th Canadian Conference on Earthquake Engineering (9NCEE)*. (47)

- RENZ, J. & LIGOZAT, G. (2005). Weak composition for qualitative spatial and temporal reasoning. In P. van Beek, ed., *Principles and Practice of Constraint Programming - CP 2005*, vol. 3709 of *Lecture Notes in Computer Science*, 534–548, Springer Berlin / Heidelberg. (22)
- RENZ, J. & MITRA, D. (2004). Qualitative direction calculi with arbitrary granularity. In C. Zhang, H.W. Guesgen & W.K. Yeap, eds., *Proceedings of the 8th Pacific Rim International Conference on Artificial Intelligence: Trends in Artificial Intelligence, PRICAI 2004*, vol. 3157 of *Lecture Notes in Computer Science*, 65–74, Springer Berlin / Heidelberg. (21, 22)
- RENZ, J. & NEBEL, B. (2007). Qualitative spatial reasoning using constraint calculi. In M. Aiello, I. Pratt-Hartmann & J. van Benthem, eds., *Handbook of Spatial Logics*, 161–215, Springer. (22, 39)
- RIGAUX, P. & SCHOLL, M. (1995). Multi-scale partitions: Application to spatial and statistical databases. In M.J. Egenhofer & J.R. Herring, eds., *Proceedings of the 4th International Symposium on Advances in Spatial Databases, SSD'95*, vol. 951 of *Lecture Notes in Computer Science*, 170–183, Springer. (14, 73, 181)
- RIGAUX, P., SCHOLL, M. & VOISARD, A. (2002). *Spatial databases with application to GIS*. Morgan Kaufmann. (12, 14, 73, 181)
- ROBINSON, V.B. & FRANK, A.U. (1985). About different kinds of uncertainty in collections of spatial data. In *Proceedings of the Seventh International Symposium on Computer-Assisted Cartography (Auto-Carto 7)*, 440–449. (22, 41, 42)
- ROY, A.J. & STELL, J.G. (2001). Spatial relations between indeterminate regions. *International Journal of Approximate Reasoning*, **27**(3), 205–234. (44, 63)
- SCHLIEDER, C. (1995). Reasoning about ordering. In A. Frank & W. Kuhn, eds., *Spatial Information Theory A Theoretical Basis for GIS*, vol. 988 of *Lecture Notes in Computer Science*, 341–349, Springer Berlin / Heidelberg. (21, 22)
- SCHNEIDER, M. (1999). Uncertainty management for spatial data in databases: Fuzzy spatial data types. In R. Güting, D. Papadias & F. Lochovsky, eds., *Advances in Spatial Databases*, vol. 1651 of *Lecture Notes in Computer Science*, 330–351, Springer Berlin / Heidelberg. (42, 43, 44)
- SCHNEIDER, M. (2002). Implementing topological predicates for complex regions. In *Proceedings of the International Symposium on Spatial Data Handling*, 313–328. (40, 125)
- SCHOCKAERT, S., CORNELLS, C., DE COCK, M. & KERRE, E.E. (2006). Fuzzy Spatial Relations between Vague Regions. In *Proceedings of the 3rd International IEEE Conference on Intelligent Systems*, 221–226. (44)
- SCIVOS, A. & NEBEL, B. (2001). Double-crossing: Decidability and computational complexity of a qualitative calculus for navigation. In D. Montello, ed., *Spatial Information Theory*, vol. 2205 of *Lecture Notes in Computer Science*, 431–446, Springer Berlin / Heidelberg. (25)
- SHAN-XIN, Z. & RUI-LIAN, Q. (2010). An algorithm for computing the union, intersection and difference of two polygons. In *Proceedings of the 2010 Second International Conference on Computer Research and Development, ICCRD '10*, 344–348, IEEE Computer Society. (17)

- SHARMA, J. (1996). *Integrated spatial reasoning in geographic information systems: Combining topology and direction*. Ph.D. thesis, University of Maine. (39, 146)
- SKIADOPOULOS, S. & KOUBARAKIS, M. (2004). Composing cardinal direction relations. *Artificial Intelligence*, **152**(2), 143–171. (13, 31, 33, 34, 44, 76, 104, 116)
- SKIADOPOULOS, S. & KOUBARAKIS, M. (2005). On the consistency of cardinal direction constraints. *Artificial Intelligence*, **163**(1), 91–135. (35)
- SKIADOPOULOS, S., GIANNOUKOS, C., VASSILIADIS, P., SELLIS, T.K. & KOUBARAKIS, M. (2004). Computing and handling cardinal direction information. In E. Bertino, S. Christodoulakis, D. Plexousakis, V. Christophides, M. Koubarakis, K. Böhm & E. Ferrari, eds., *Advances in Database Technology - EDBT 2004, Proceedings of the 9th International Conference on Extending Database Technology*, vol. 2992 of *Lecture Notes in Computer Science*, 329–347, Springer. (40)
- SKIADOPOULOS, S., GIANNOUKOS, C., SARKAS, N., VASSILIADIS, P., SELLIS, T. & KOUBARAKIS, M. (2005). Computing and managing cardinal direction relations. *IEEE Transactions on Knowledge and Data Engineering*, **17**(12), 1610–1623. (40)
- STEINHAUER, H.J. (2008). *A Representation Scheme for Description and Reconstruction of Object Configurations Based on Qualitative Relations*. Ph.D. thesis, Linköping University Electronic Press. (71)
- STEINIGER, S. & BOCHER, E. (2009). An overview on current free and open source desktop GIS developments. *International Journal of Geographical Information Science*, **23**(10), 1345–1370. (149)
- TARQUINI, F., DE FELICE, G., FOGLIARONI, P. & CLEMENTINI, E. (2007). A qualitative model for visibility relations. In J. Hertzberg, M. Beetz & R. Englert, eds., *Proceedings of the 30th Annual German Conference on Advances in Artificial Intelligence*, vol. 4667 of *Lecture Notes in Computer Science*, 510–513, Springer Berlin / Heidelberg. (21, 36, 37, 138)
- TASSONI, S., FOLIARONI, P., BHATT, M. & DE FELICE, G. (2011). Toward a qualitative model of 3d visibility. In *25th International Workshop on Qualitative Reasoning, IJCAI 2011*, (position paper). (21)
- TØSSEBRO, E. & NYGÅRD, M. (2008). Representing Uncertainty in Spatial Databases. In *High Performance Computing and Simulation Conference*, 141–152. (43, 44)
- TYE, M. (1990). Vague objects. *Mind*, **99**(396), 535–557. (42)
- VARZI, A.C. (2001). Vagueness in geography. *Philosophy & Geography*, **4**(1), 49–65. (42)
- VARZI, A.C. (2006). Vagueness. In *Encyclopedia of Cognitive Science*, John Wiley & Sons, Ltd. (42)
- WALLGRÜN, J.O., FROMMBERGER, L., WOLTER, D., DYLLA, F. & FREKSA, C. (2007). Qualitative spatial representation and reasoning in the SparQ-toolbox. In T. Barkowsky, M. Knauff, G. Ligozat & D. Montello, eds., *Spatial Cognition V: Reasoning, Action, Interaction*, vol. 4387 of *Lecture Notes in Computer Science*, 39–58, Springer. (106, 131)

- WANG, M. & HAO, Z. (2010). Reasoning about the Inverse of Cardinal Direction Relation. *Information Technology Journal*, **9**(9), 935–941. (35)
- WEIN, R. (2007). Exact and approximate construction of offset polygons. *Computer-Aided Design*, **39**(6), 518–527. (16)
- WOLTER, D. & WALLGRÜN, J.O. (2012). Qualitative spatial reasoning for applications: New challenges and the sparq toolbox. In S.M. Hazarika, ed., *Qualitative Spatio-Temporal Representation and Reasoning: Trends and Future Directions*, 336–362, IGI Global. (40, 71, 73, 101, 104)
- WORBOYS, M. (1998a). Computation with imprecise geospatial data. *Computers, Environment and Urban Systems*, **22**(2), 85–106. (41)
- WORBOYS, M. (1998b). Imprecision in finite resolution spatial data. *GeoInformatica*, **2**(3), 257–279. (41, 42, 43)
- WORBOYS, M. & DUCKHAM, M. (2004). *GIS: A Computing Perspective, 2nd Edition*. CRC Press, Inc. (12)
- ZADEH, L.A. (1965). Fuzzy sets. *Information and Control*, **8**(3), 338–353. (42)
- ZADEH, L.A. (1979). Fuzzy sets. In A. Holzman, ed., *Operations Research Support Methodology*, 569–606, Marcel Dekker, New York. (42)
- ZADEH, L.A. (1995). Probability theory and fuzzy logic are complementary rather than competitive. *Technometrics*, **37**(3), 271–276. (42)
- ZHAN, F.B. (1998). Approximate analysis of binary topological relations between geographic regions with indeterminate boundaries. *Soft Computing*, **2**(2), 28–34. (42, 44)
- ZHU, B., DAI, X. & LI, X. (2010). Composing fuzzy cardinal direction relations. In F. Sun, Y. Wang, J. Lu, B. Zhang, W. Kinsner & L.A. Zadeh, eds., *9th IEEE International Conference on Cognitive Informatics (ICCI 2010)*, 777–782. (44)
- ZHU, C., SUNDARAM, G., SNOEYINK, J. & MITCHELL, J.S.B. (1996). Generating random polygons with given vertices. *Computational Geometry Theory and Applications*, **6**(5), 277–290. (155)
- ZIMMERMANN, K. (1995). Measuring without measures: the delta calculus. In W.K. A Frank, ed., *Spatial Information Theory: a Theoretical Basis for GIS*, vol. 988 of *Lecture Notes in Computer Science*, 59–68, Springer Verlag, Berlin. (21)
- ZOOK, M., GRAHAM, M., SHELTON, T. & GORMAN, S. (2010). Volunteered geographic information and crowdsourcing disaster relief: A case study of the haitian earthquake. *World Medical Health Policy*, **2**(2), 7–33. (52)