Qualitative Spatial Reasoning for Applications: New Challenges and the SparQ Toolbox

Diedrich Wolter* Jan Oliver Wallgrün†

Abstract

About two decades ago the field of qualitative spatial and temporal reasoning (QSTR) has emerged as a new area of AI research that set out to grasp human-level understanding and reasoning about spatial and temporal entities, linking formal approaches to cognitive theories. Empowering artificial agents with QSTR capabilities is claimed to facilitate manifold applications, including robot navigation, geographic information systems (GIS), natural language understanding and computer-aided design. QSTR is an active field of research that has developed many representation and reasoning approaches so far, but only comparatively few applications exist that actually build on these QSTR techniques.

In this article we approach QSTR from an application perspective. Considering the exemplary application domains of robot navigation, GIS, and computer-aided design, we conclude that reasoning must be interpreted in a broader sense than the often-considered constraint-based reasoning and that supporting tools must become available. We discuss the newly identified reasoning tasks and how they can be supported by QSTR toolboxes to foster the dissemination of QSTR in applications. Furthermore, we explain how we aim to overcome the lack-of-tools dilemma through the development of our QSTR toolbox SparQ.

1 Introduction

Qualitative spatial and temporal reasoning (QSTR) (Cohn & Hazarika, 2001; Cohn & Renz, 2007; Renz & Nebel, 2007) is the subfield of knowledge representation and symbolic reasoning that deals with knowledge about an infinite spatio-temporal domain using a finite set of *qualitative relations*. One particular aim is to model human common-sense understanding of space. Qualitative approaches have therefore been promoted as a basis for connecting human cognition and intelligent agents. Moreover, qualitative approaches offer compact representations that are supposed to enable complex decision tasks. However, despite these rationales, we still observe a lack of success stories of QSTR in the

*SFB/TR 8, Universität Bremen
†IRTG GRK 1498, Universität Bremen

sense of successful applications that make use of QSTR, or, ultimately, applications that are successful *because* they make use of QSTR.

In this article, we explore qualitative reasoning from an application oriented point of view. One possible reason for the comparatively small number of QSTR applications could be seen in a lack of adequate software toolboxes which provide the results of QSTR research in a form which enables application developers to incorporate QSTR techniques easily into their own software. Throughout the last years, first QSTR toolboxes have started to emerge (e.g., GQR (Gantner, Westphal, & Wölfl, 2008), QAT (Condotta, Ligozat, & Saade, 2006), and our own toolbox SparQ (Wallgrün, Frommberger, Wolter, Dylla, & Freksa, 2007)). However, so far these efforts have been very much concentrated on the problem of deciding satisfiability of sets of qualitative constraints. This emphasis on what we will term *constraint-based reasoning* is easily understandable because, technically speaking, qualitative relations constrain the valuation of variables and, hence, deciding satisfiability has been in the center of theoretical research in QSTR during the last two decades. Contrary to classical constraint-based techniques, QSTR has pursued a purely relation algebraic approach: qualitative relations and operations on them constitute a qualitative calculus (Ligozat & Renz, 2004) and the operations provide a symbolic approach to deciding consistency (Renz & Nebel, 2007).[1]

One thesis underlying this work is that qualitative reasoning goes beyond constraint-based reasoning and, hence, different forms of reasoning need to be supported by the toolboxes. To corroborate this claim we look at three potential application domains of QSTR, namely the areas of robot navigation, geographic information systems (GIS), and computer-aided design. Our goal is to identify which kind of qualitative reasoning is required to solve the individual problems of spatial knowledge processing occurring in these domains and our conclusion will be that constraint-based satisfiability testing often only plays a minor role.

Based on this result, we organize the newly identified reasoning tasks into groups and discuss how well they are currently understood, how they need to be supported in QSTR toolboxes, and what kind of theoretical research is still required. We then provide a glimpse at our own QSTR toolbox SparQ which we are developing with the goal of providing an easy-to-use interface to a rich repository of qualitative calculi and reasoning methods. In particular, we describe to which extent SparQ already supports some of the newly identified reasoning tasks and sketch its future development.

While the focus of this article will be on different kinds of reasoning tasks where reasoning is interpreted in a rather broad sense, there are clearly other issues involved that would improve the dissemination of QSTR techniques and toolboxes, e.g., representational aspects and integration with other AI methodology. We will address some of these points in the final section of this text containing conclusions and an outlook.

The article is organized as follows. In Section 2 we look at the different kinds

[1] A general, unspoken assumption underlying many QSTR calculi is that algebraic closure decides consistency for constraint networks involving base relations only (cp. (Renz, 2007); this is not the case in general though (Lücke, Mossakowski, & Wolter, 2008)).

of reasoning tasks occurring in the previously mentioned application domains. In Section 3 we classify and discuss the newly identified reasoning tasks. Our own toolbox SparQ will then be presented in Section 4, followed by conclusions and an outlook in Section 5.

2 Three Application Areas for QSTR

Three application domains which are often cited as natural application areas for qualitative spatial reasoning are robot navigation, geographic information systems (GIS), and computer-aided design (see, among others Cohn & Hazarika, 2001; Cohn & Renz, 2007; Egenhofer & Mark, 1995; Bhatt, Dylla, & Hois, 2009). We will see that distinct spatial reasoning tasks can be identified and most of them appear throughout all of the three application areas.

2.1 QSTR and Mobile Robot Navigation

Being able to move through the environment and successfully reach a particular goal location is a fundamental ability for mobile robots as they are currently being developed for, e.g., service or autonomous exploration tasks. Navigation essentially is concerned with three questions (Levitt & Lawton, 1990): "Where am I?", "Where are other places relative to me?", and "How do I get to other places from here?". Two important problems distinguished in robotics are *self-localization* (determining one's position wrt. to an internal representation of the environment) and *map learning* (autonomously acquiring such an internal model of an initially unknown environment). In the following we look at these tasks individually. We also examine the emerging application field of language-based *human-robot interaction* where one would like to be able to communicate with robots using spatial concepts and natural language descriptions.

2.1.1 Self-Localization

For goal-directed navigation, a robot needs to be able to localize itself within its own spatial representation of the environment, also termed its *map*. This task requires the robot to recognize locations in the map from observations made within the environment. Two principle approaches have been pursued: searching for a robot location within the map that would explain the robot's observation best (see, e.g., Fox, Burgard, & Thrun, 1999) or searching the map for the features observed (see, e.g., Castellanos, Neira, & Tardós, 2006). In the following we subscribe to the feature-based approach which makes self-localization a two-step process: First, the correct correspondences between objects currently perceived by the robot and objects in the map need to be established. The problem of finding the correct associations is often referred to as the *correspondence problem* or the *data association problem* (Bar-Shalom & Fortmann, 1988; Grimson, 1990; Neira & Tardós, 2001). Second, the location of the robot is determined based on its spatial relation with respect to the perceived objects.

Evaluating all potential correspondences between the objects observed and those stored in the map easily results in combinatorial explosion, so it can be beneficial to consider qualitative spatial relations among the objects to constrain data association, thereby enhancing efficiency (Wallgrün, 2010; Wolter, 2008). Qualitative relations constitute robust relations that are insensitive to measurement noise and, consequentially, can be utilized as hard constraints to restrict the search space of possible correspondences. An example for such a relation can be "left of" (Wolter, Freksa, & Latecki, 2008), or landmark panoramas (Schlieder, 1993) based on cyclic order. Exploiting qualitative constraints in robot self-localization comprises four tasks that are illustrated in Figure 1: First, the relations between all objects in the map are interpreted in terms of qualitative spatial relations (this only needs to be done once) and, in the second step, the same is done with the local observation. We obtain two qualitative constraint networks which are represented as graphs, one representing the map, and the other one representing the observation. In the third step the correspondence problem would be tackled by finding a subgraph of the map isomorphic to the one obtained from the observation such that both graphs share the same constraint labels. However, one has to take into account that we might perceive objects for the first time and as a result they will not be contained in the map yet. Hence, we actually are looking for the largest common subgraph of the constraint graphs generated from observation and map which, under general assumptions, yields the most likely matchings between observation and map.

2.1.2 Map Learning

The task of autonomously acquiring a map of the environment is called map learning, robot mapping, or the simultaneous localization and mapping (SLAM) problem (Thrun, 2002; Leonard & Durrant-Whyte, 1991; Castellanos et al., 2006). A significant part of map learning of course is localization as discussed above in order to allow the robot to register new parts of the environment at the right position with respect to the parts already mapped. The resulting map will only be correct if the data association decisions made are correct. However, one cannot assume that the correct associations are made in every localization performed during the mapping process. One approach to deal with errors in the data associations is to retract previous decisions when they lead to a map representation that is spatially inconsistent (Hähnel, Burgard, Wegbreit, & Thrun, 2003). Alternatively, different map hypotheses, each based on a particular history of data association decisions, can be tracked and maintained simultaneously (Dudek, Freedman, & Hadjres, 1996; Kuipers, Modayil, Beeson, MacMahon, & Savelli, 2004). Whenever one of the map hypotheses becomes inconsistent, that is the perceived spatial relations holding between the objects cannot be satisfied for the data association decisions made, this hypothesis can be discarded. However, for both approaches it is crucial that the information on which the decision to discard a map hypothesis is based can be assumed to be perceivable reliably. For instance, in the approach tracking multiple hypotheses, we would want to make sure that we do not discard the correct hypothesis

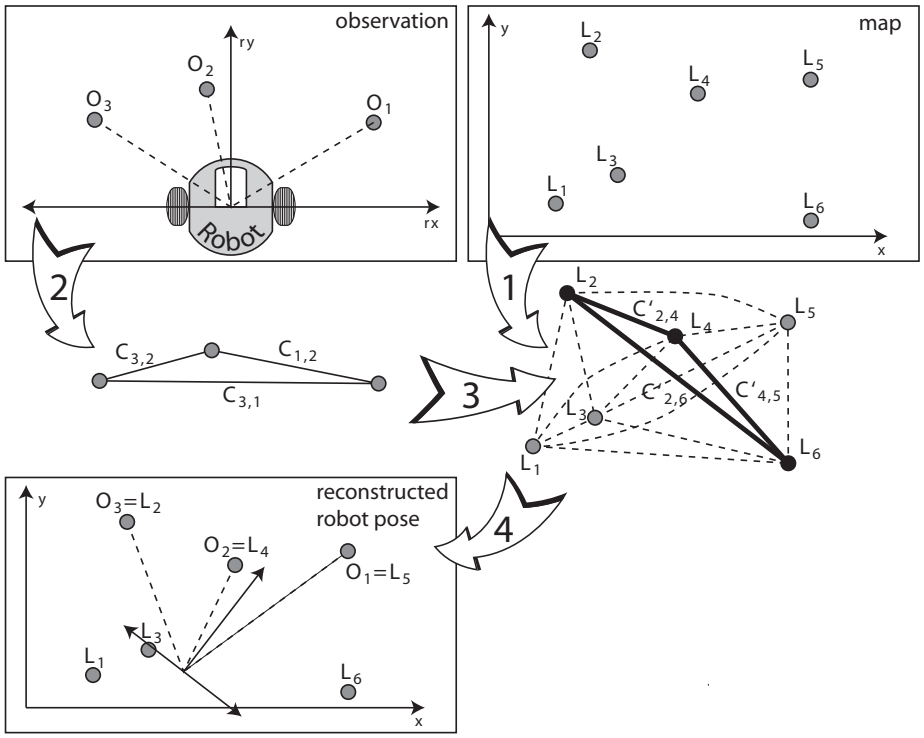


Figure 1: Four steps in exploiting qualitative relations in robot-self-localization

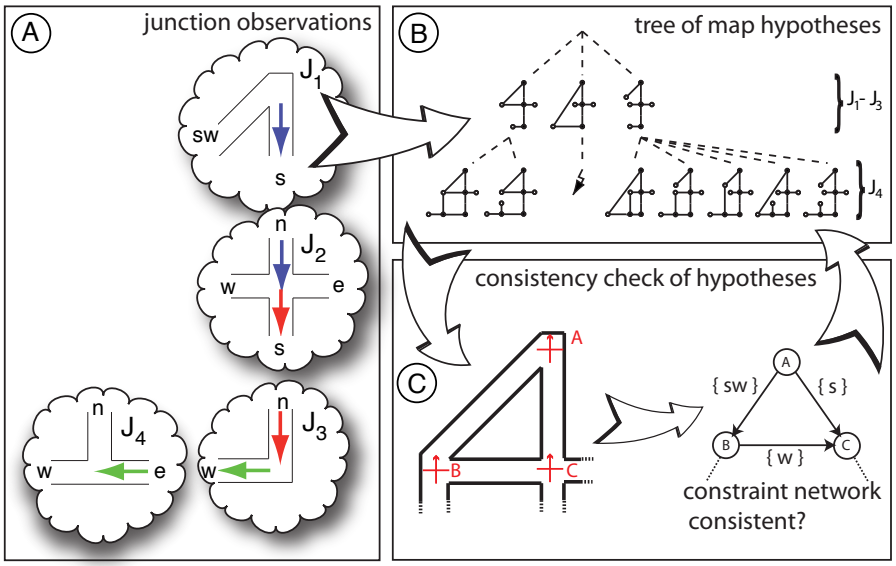


Figure 2: Topological map learning using qualitative direction information and consistency checking

because a precise metric relation does not hold due to sensor noise. This is where map learning can benefit from employing qualitative relations and, as a result, QSTR approaches for deciding the consistency of a map hypothesis are valuable.

One example for the described way of applying QSTR for map learning can be found in (Wallgrün, 2009). The approach extends earlier work of Moratz and Wallgrün (2003) and Moratz, Nebel, and Freksa (2003) and aims at learning a particular kind of spatial representation called a topological map (Remolina & Kuipers, 2004; Kuipers, 2000) from coarse but reliably perceivable spatial relations (see Figure 2 for an illustration). A topological map is a graph-based representation in which the nodes stand for relevant places (e.g., junctions) and the edges connect adjacent places (e.g. they stand for hallways). The exploration history of the robot consists of junction observations and hallway traversal actions (Figure 2 (A)). The perceived directions of leaving hallways are mapped to qualitative direction relations from a qualitative direction calculus which can then be treated as reliable information. Instead of maintaining a single map, multiple map hypotheses are tracked simultaneously as illustrated in Figure 2 (B). This results in a search tree of map hypotheses that represent different data association decisions made. Consistency checking based on the direction information contained in the individual hypotheses using standard QSTR techniques is then employed to discard invalid hypotheses (Figure 2 (C)).

Approaches in which qualitative spatial relations are used as reliably perceivable constraints but for representations based on landmarks have for instance

been discussed in (Moratz, 2009; Steinhauer, 2008).

2.1.3 Human-Robot Communication

One application researched in the field of robotics is to develop service robots which co-exist within a human society. Service robots are supposed to take orders from humans and, in some cases, report back to humans, or request more information to resolve ambiguities. In these scenarios, being able to communicate spatial information is a key capability. Among the multitude of modalities that can be employed in a human-robot interface, natural language is possibly the most desirable one. In order to interpret spatial information captured in natural language expressions, formal models of spatial relations and reference systems are required—the ability to handle relative spatial information is of great importance (Moratz & Tenbrink, 2006). In order to interpret relative spatial information it is necessary to change the frame of reference accordingly, e.g., "to the left" from the instructor's perspective translates "to the right" for an opposing robot. Such perspective changes can be realized with the help of the permutation operations (e.g., converse) defined in a qualitative calculus. We refer to these simple reasoning steps as *computing with relations*.

Considering the ability of a robot to communicate spatial information to a human, the use of qualitative relations is also advantageous. If we consider route descriptions that incrementally relate landmarks as, for example, "after turning right at the intersection the hardware store is to the left", constraint-based qualitative reasoning techniques become necessary to integrate information relative to path segments into a coherent whole. An example of using path integration of spatial knowledge for robot instruction can be found in Krieg-Brückner and Shi (2006). Ultimately, methods are required that allow a robot to find its way and recognize the goal position it has been requested to move to purely based on natural language expression. Thus, robot instruction involves a generalized form of self-localization that localizes the robot with respect to a location description (cp. Wolter et al., 2008).

2.2 QSTR and GIS

Another application area well-suited for qualitative representations and reasoning is that of geographic information systems (GIS) which we approach in a general sense such as to comprise spatial data infrastructures, location-based service, etc. A typical GIS provides means to store, update, query, analyze, and visualize geospatial information (Burrough, 1986). A typical modern GIS stores spatial information quantitatively using either a vector-based approach or a raster-based approach (Winter, 1998). In the vector-based approach, different kind of geometries (line, point, polygon, etc.) can be used to describe the spatial extension of each object. There exist, however, several reasons why one would like to extend traditional GIS with the ability to directly store and process qualitative spatial relations. For instance, one could have one of the following situations:

- There is only qualitative information available, for instance innformation stemming from a sensor that can only distinguish certain qualities.
- Information is provided by a human—being able to use qualitative relations in queries or when feeding information to the system makes the interface more convenient and intuitive.
- Only imprecise information shall be provided about a location due to privacy concerns (Duckham & Kulik, 2005).
- Only the qualitative information can be assumed to be reliable, for instance when the information is extracted from a sketch map (Lovett, Dehghani, & Forbus, 2007).
- Only distinguishing categories that make a difference will increase the efficiency of an analysis or the spatial reasoning that needs to be performed.

Qualitative spatial representation and reasoning formalisms have always been of interest in geographic information science following the idea of a naive geography (Egenhofer & Mark, 1995) which is concerned with formal models of human common-sense reasoning about geographic space. While there exists a significant amount of theoretical research, for instance regarding consistency of topological relations in a multi-layer representation (Egenhofer, Clementini, & Felice, 1994; Belussi, Catania, & Podestà, 2005), very little of this work has found its way into existing GISs. Support for qualitative spatial representation and reasoning here is still mostly limited to realizing a set of topological predicates based on a topological formalism like the 9-intersection model (Egenhofer, 1989) or RCC-8 (Randell, Cui, & Cohn, 1992), e.g., relations like contains, overlaps, etc., as extensions of query languages such as SQL (Egenhofer, 1994; OpenGIS Consortium, 1999). When used for querying, the qualitative relations holding between objects are computed from the objects' geometries every time they are needed.

In the following we will look more closely at what kinds of demands for qualitative reasoning and representations methods arise from different subareas of the GIS domain. Although temporal aspects are also of great importance in GIS, we focus on aspects of spatial reasoning in our discussion. The identified tasks and approaches can in most cases be directly transferred to the temporal domain.

2.2.1 Query Processing

In many application scenarios, it is more natural for human users to pose queries in a qualitative way, e.g., by using a natural language interface (see Figure 3). A natural language query such as "give me all camping sites north of lake Constance which are directly at the lakeside but also close to a town" first has to be formalized using spatial relations from a qualitative calculus (cmp. Section 2.1.3). The result can be seen as a constraint network Q with three objects C (camping site), L (lake), T (town) and the constraints:

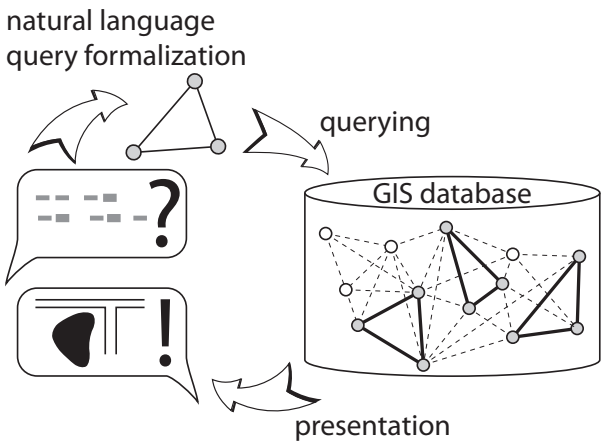


Figure 3: Three steps in querying GIS databases qualitatively

$$C\ (\texttt{north\_of}, \texttt{externally\_connected\_to}, *)\ L$$

$$C\ (*, *, \texttt{close\_to})\ T$$

The constraints here are triples of relations from the three individual calculi used in the query: a cardinal direction calculus, a topological calculus based on RCC-8, and a proximity calculus. The '*' stands for the universal relations of the respective calculus. In an information system in which this kind of queries are common, it makes sense to store qualitative relations explicitly to avoid the additional costs of having to compute the qualitative information from the geometries every time and to provide special data structures in order to increase the efficiency of query processing. Hence, we can regard the spatial database as a large constraint network D in which the constraints consist of relations from several qualitative calculi. Query answering can then be seen as the subgraph isomorphism problem of finding all occurences of Q in D. This task is similar to what we discussed in the Section 2.1 regarding localization except that here we are looking for complete instances of Q rather than largest subgraphs of Q in D.

2.2.2 Similarity Assessment

The description in the previous section assumes that the result of a query is given in form of all instances that are subsumed by the query. However, often one would prefer to query by example and look for instances that are most similar to the query. For querying spatial configurations, similarity could involve several things: similarity between the concepts involved, similarity of object instance, or the similarity of the spatial relations holding between the objects. The problem of assessing the semantic similarity between concepts or objects has recently received increasing attention (Rodríguez & Egenhofer, 2004; Schwering,

2008; Janowicz, Raubal, Schwering, & Kuhn, 2008). With regard to qualitative spatial relations, one idea is to assess the similarity of two relations based on the notion of conceptual neighborhood (Freksa, 1991, 1992) by considering the distance between base relations in the conceptual neighborhood graph (Bruns & Egenhofer, 1996; Dylla & Wallgrün, 2007). The concrete conceptual neighborhood structure depends on the concrete set of continuous transformations we assume (Freksa, 1991; Dylla & Wallgrün, 2007) which in turn need to be grounded in spatial change over time (Galton, 2000). This approach can be extended to measure the similarity of complete spatial configurations which would allow to find the most similar spatial configuration to a given configuration or to generate similar configurations to a given one. As an overall goal, one would like to integrate techniques for assessing the different kinds of similarities (concept-concept, instance-instance, relation-relation) into a single general framework able to take all these aspects of similarity into account when processing a query.

2.2.3 Data Integration

The problem of data integration occurs in many GIS applications (Fonseca, Egenhofer, Agouris, & Câmara, 2002; Duckham & Worboys, 2005; Bittner, Donnelly, & Smith, 2009). Databases may need to be merged to create a new single knowledge base or data from different spatial data infrastructures should be combined on the fly in order to solve a particular task. Typically, the integration task comprises several subproblems:

- semantic alignment of the involved concepts
- identifying corresponding entities across different data sets
- determining conflicts between the different data sets
- resolving or managing these conflicts in a suitable way
- merging the data into a single representation

Focusing on the processing of qualitative knowledge required to solve these subproblems, we can identify several crucial issues: When combining information from different sources, we might be faced with the problem of having to merge qualitative information given in terms of different spatial calculi. To have a theoretical basis for this merging step, we need to know how the relations from different calculi are related with each other and how they constrain each other. In short: a combined calculus is needed to make sure that no information is lost and that all potential conflicts are discovered. It would be desirable if a suitable combined calculus could be constructed automatically based on a semantic description of the relations of the involved calculi. The result would again be the semantic descriptions of the relations of the combined calculus together with relevant operations like the composition. On the other hand, when we are combining information that is partially qualitative and partially quantitative, the

quantitative information must first be transferred to the qualitative level before the real combination step can take place.

Automatically determining the correspondences between the databases to be merged poses the same problem we discussed in the localization scenario, namely finding a largest common subgraph between the databases which is isomorphic. This subgraph would then yield the correspondences between objects contained in both data sets and the objects which are only contained in one of them. However, unless the conceptual information available about the objects in the databases is very restrictive, it is unlikely that an automatic process will be able to always determine the correct correspondences. Hence, we imagine this will rather be a semi-automatic process in which a user verifies and corrects the result of the automatic matching.

Once the gaps on the qualitative level are filled and the correspondences are established, the information can be combined. Combining can be seen as the task of merging qualitative constraint networks. However, it is possible that the information to be merged is conflicting and, hence, simply taking the intersections over all constraints will result in an inconsistent constraint network. Discovering these conflicts is an application of standard constraint-based reasoning. However, in many applications one would like to also be able to resolve conflicts that occur by relaxing the constraints in the original data sets in a meaningful way (Dylla & Wallgrün, 2007; Condotta, Kaci, & Schwind, 2008). For instance, a common-sense approach would be to look for constraint networks which are most similar to the original networks but yield a consistent network when they are combined. As discussed in Section 2.2.2, a notion of similarity between constraint networks can be derived based on the distance of base relations in a conceptual neighborhood graph.

2.2.4 Spatial Analysis

GISs provide many different ways to analyze spatial data and processes including explanation and interpolation, simulation and prediction, and planning (Smith, Goodchild, & Longley, 2007). Performing these tasks on a qualitative level, it is possible to only consider the distinctions that make a difference. This increases the efficiency and simplifies the interpretation of the results by humans (Kuipers, 1994). The basic component that is needed is a way to describe potential transitions between qualitative configurations of more than two objects. As discussed in Section 2.2.2, conceptual neighborhood distance can be used to achieve this. The result is a generalized neighborhood graph as described by Ragni and Wölfl (2005) in which the nodes stand for consistent scenarios and the edges connect conceptually neighbored scenarios.

Given such a generalized neighborhood graph or a way to generate the adjacent scenarios for a given scenario, different possible temporal developments can be predicted and simulated straightforwardly. An example of this general approach is the work described in Cohn et al. (1997) about simulating phagocytosis and exocytosis in unicellular organisms using the RCC-8 relations.

In a similar way explanation of spatial processes can be realized via interpo-

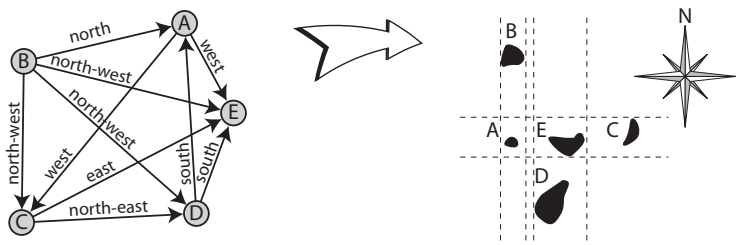


Figure 4: Quantification determines a model for a qualitative constraint network that can be used for visualization

lation between a set of static snapshots which could explain what has happened in between. Interpolating between two spatial snapshots—which we again can imagine as constraint networks—would mean to find a sequence of scenarios which leads from the start scenario to the goal scenario. Given a generalized conceptual neighborhood graph, we can look for the shortest path connecting the snapshots as the "simplest" explanation of what could have occurred. The same approach can be employed for planning where the snapshots are replaced by start and goal scenarios (Ragni & Wölfl, 2006).

2.2.5 Visualization

A main task of a GIS is providing flexible ways to visualize the stored data and the intermediate steps during a spatial analysis (Hearnshaw & Unwin, 1994). For most kind of geometric data, visualization is unproblematic and research is more focused on how to adapt the visualization to meet the implicit demands of the user. For qualitative information the situation is more complicated but nevertheless we want to be able to produce visualizations even when parts or all information is given qualitatively. As an example application, let us consider a location-based system which accepts inputs about the user's location relative to some prominent buildings in the following form: "I am facing the cathedral and there is a fountain to my left and a street to my right". The location-based system is supposed to generate a you-are-here-map for the qualitatively described position of the user.

Applications of this kind require a way to derive new information from a mixture of quantitative and qualitative information and to generate a quantitative scene description (e.g., a drawing or a map) from a qualitative scene description or even from a mixed qualitative-quantitative scene description. A first step in this direction would be the ability to compute one exemplary quantitative solution for a qualitative scenario as shown in Figure 4. To deal with general constraint networks including disjunctions of base relations, one would then need to extend this approach, for instance by presenting one exemplary solution for each consistent qualitative scenario to the user or by providing ways to navigate through the solution space.

2.3 QSTR and Computer-Aided Design

Qualitative representation and reasoning can be important in the area of computer assistance in design processes, particularly by providing means to handle design constraints. Concrete tasks in computer-aided design that involve QSTR range from formal specification of design constraints to model checking and construction of solutions that are admissible with respect to the design constraints. Qualitative relations are valuable from on an early stage of engineering and architectural design processes, as they enable representation of uncertain knowledge which is common in early stages of a design (Schultz, Amor, Lobb, & Guesgen, 2009). Based on a formalization of design constraints, model-checking can be performed with (partially) existing designs; qualitative methods would be used in conjunction with a general logical formalism, see, e.g., (Bhatt et al., 2009). In situations where designs are not acceptable, methods can be applied that resolve conflicting constraint networks and that externalize qualitative knowledge. A qualitative design-support system would be able to resolve conflicts and to suggest alternative solutions. Some realizations of the ideas sketched above can already be found in the work of Richter, Weber, Bojduj, and Bertel (to appear). Furthermore, qualitative knowledge helps to assess the complexity of building layouts from a wayfinder's point of view (Bojduj, Weber, Richter, & Bertel, 2008) which then can be used to design environments that ease wayfinding tasks.

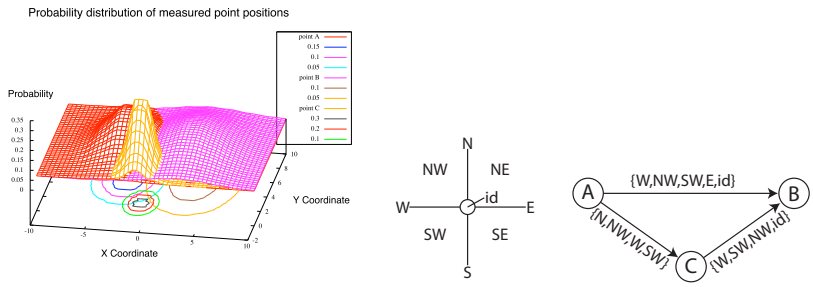
Summing up, there are several QSTR reasoning tasks can provide assistance in design processes. For checking the consistency of a design with respect to design constraints, quantitative data has to be interpreted qualitatively (qualification) and constraint network have to be compared. Given that some design constraints are not met, relaxation of conflicting constraints helps to construct an admissible design on the qualitative level. By graphically externalizing the qualitative information (quantification), concrete design recommendations can be made in an easily comprehensible way.

3 QSTR Support for Applications

In our inspection of the three application areas in the previous section, we identified a large potential for qualitative representation and reasoning approaches. However, several services required in these applications were not directly based on traditional constraint-based reasoning which has been the focus of theoretical research and development of reasoning software over the last two decades. In this section, we summarize and structure the identified services by dividing them into six groups.

3.1 Qualification

As we have seen, many applications in which qualitative spatial information is processed, first require that data is transferred from the quantitative world (e.g., a geometric scene description) to the qualitative world. We will refer to this



(a) (b) (c)

Figure 5: Mapping noisy data to qualitative information. (a) probability distribution of measured point position, (b), cardinal direction relations, and (c) possible interpretation into qualitative information

process of computing qualitative relations from quantitative data as *qualification*. Qualification always occurs with regard to a given spatial calculus defining the available set of relations. A software module performing qualification for a particular calculus $\mathcal{C}$ will be called a *qualifier for* $\mathcal{C}$.

Typically, one would expect to obtain a consistent scenario as the result of qualification, i.e., the qualitative scene description would contain only base relations of $\mathcal{C}$ which hold between the objects and the obtained constraint network is consistent. However, this task is challenging in two regards.

First, unavoidable rounding errors which occur in processing floating point numbers complicate the interpretation of data provided as floating point values. Floating point values discretize a continuous domain. As a consequence, for calculi which define exact relations like point coinciding with lines, it may be impossible to decide whether the data supplied is *meant* to be related by an exact relation. Thus, mapping floating point values to qualitative labels may introduce misinterpretations and, henceforth, the obtained constraint network may be inconsistent. It remains an open problem to efficiently determine a consistent qualitative constraint network only using atomic relations such that the constraint network optimally fits to the supplied data.

Second, any data obtained from sensors is affected by measurement noise. Thus, we cannot interpret the data as true information but we must also consider possible deviations of the values. As a consequence, it may not be possible to interpret object relationships in terms of base relations. Instead, we obtain a constraint network that includes (some) disjunctive relations. We illustrate the effects of measurement noise in an example using the cardinal direction calculus (Frank, 1991; Ligozat, 1998) depicted in Figure 5. In the figure we can observe an actual challenge of qualifying data that is subject to noise: devising an efficient algorithm that enumerates all base relations that can hold between the objects. Difficulty arises from the need to evaluate all values from a continuous, hence, infinite set of values.

To our knowledge, these issues have not been tackled so far. If we do not con-

sider noise in the input data, implementing a specific qualifier is rather straightforward. We note that no actual programming is required if the qualitative relations can be specified in a formal framework—a unified qualifier module can then match geometric scene descriptions to the formal specifications, identifying which relation holds. For example, for calculi over points in the plane or more generally in n-dimensional Euclidean space it is typically sufficient to check a set of criteria given in terms of (in)equality equations for each base relation (e.g., Euclidean distance of points or angles). In conclusion, providing a reasonable qualifier is not as trivial as it might look at first glance.

3.2 Manipulation, Retrieval, and Comparison

Qualitative relations are useful for formalizing queries in, for example, GIS applications. Technically, a query is given by a constraint network and one needs to compute a mapping from the variables in the query network to the variables in the database such that corresponding relations are identical. Reasoning plays an important role in making query answering efficient (Grimson, 1990). Efficient algorithms require specialized data structures. To avoid loosing time by setting up the data structures from a representation of constraint networks used in the application, it is helpful that a toolbox provides representations of constraint networks that suit efficient retrieval as well as the needs of an application. Thus, an application would not store all qualitative information itself, but it would rather utilize a constraint network representation that resides within the qualitative reasoner module used by the application. We note that the same reason holds for other qualitative reasoning tasks that involve specialized data structures. In conclusion, to avoid unnecessary setup of specialized data structures, we see the need that any qualitative reasoning toolbox provides qualitative constraint network representations that automatically update all specialized data structures. Naturally, providing a representation also requires the toolbox to offer a set of versatile operations for manipulation, access, and comparison of constraint networks. Examples of these operations are taking the intersection or union of constraint networks, accessing and changing individual constraints, and determining whether one network is a refinement of another one.

3.3 Constraint-Based Qualitative Reasoning

As qualitative relation networks are constraint networks over an infinite temporal or spatial domain, constraint-based reasoning is the typical approach to reasoning with qualitative information. The constraint-based reasoning tasks we identified are the following:

- Deciding consistency of a given network of qualitative relations
- Determining one or enumerating all consistent scenarios from a constraint network

- Removing redundancy from a constraint network, i.e., computing the minimal network (Dechter, Meiri, & Pearl, 1991)

Classically, these tasks are tackled by the algebraic closure algorithm using the composition and converse operation of a qualitative calculus. Although deciding consistency of qualitative calculi is NP-hard for almost for all calculi, this approach is very efficient (Renz, 2002). It is widely assumed that algebraic closure can be used to decide consistency of atomic constraint networks (see Renz & Nebel, 2007).

However, approaching consistency problems only by means of the algebraic closure algorithm has severe limitations. First and foremost, the assumption that algebraic closure can decide consistency of atomic network is not valid for relative orientation calculi that distinguish left- and right-hand side of arbitrary reference lines (Lücke et al., 2008). The effectiveness of algebraic closure has not been investigated for all calculi yet. Second, algebraic closure may not provide the most efficient means for deciding consistency; for example, general SAT solvers can in some situations outperform methods based on algebraic closure for constraint networks (Westphal & Wölfl, 2009). Third, any approach based on the operation tables of a calculus is confined to handling relations from one single calculus only, i.e., it is not possible to reason about networks that contain relations from different calculi. The bipath-consistency method may be used as a work-around, but this method is often too weak for deciding consistency (Wölfl & Westphal, 2009).

Additionally, we treat elementary computations performed with qualitative relations under the label of constraint-based reasoning because these tasks can in principle be posed as constraint-satisfaction problems. For example, if one wants to determine the composition of two relations r, s, one can construct a network with three arbitrary variables A, B, C and with constraints $r(A, B)$, $s(B, C)$, and the universal relation constraint between A and C. After computing the minimal network the desired composition of r and s can be read off as the relation holding between A and C. Of course, any qualitative reasoning toolbox would offer easier-to-use means that would neither require the user to set up constraint networks manually nor would the toolbox use costly CSP decision procedures if a simple relation lookup is sufficient.

3.4 Neighborhood-Based Reasoning

Under the term neighborhood-based reasoning we subsume all qualitative reasoning services which are based on the notion of conceptual neighborhood. Elementary operations we foresee here are the computation of the conceptual neighbors of a given relation and the distance of two base relations in the conceptual neighborhood graph which is defined by the length of the shortest path connecting them (Bruns & Egenhofer, 1996; Dylla & Wallgrün, 2007).

To facilitate these elementary services the conceptual neighbor relation or alternatively the conceptual neighborhood graph need to be specified as part of the calculus specification. However, the exact neighborhood structure depends

on the concrete set of transformations (move, grow, shrink, change shape, etc.) that the related objects may perform (Freksa, 1991; Dylla & Wallgrün, 2007). Hence, instead of specifying a single neighborhood structure, one would like to specify several neighborhood structures and choose then the one that best suits the application at hand.

Based on the elementary neighborhood operations, one could then realize services for assessing the similarity or distance between constraint networks and for generating potential successor scenarios of a given network. The next step are operations that perform integration tasks (including conflict resolution) or are based on some kind of search through the generalized conceptual neighborhood graph (e.g., simulation, explanation, and planning as discussed in Section 2.2). Theoretical frameworks for relaxing a single inconsistent network or for merging several networks which contradict each other by considering scenarios which are at the same time consistent and closest the original network(s) have already been developed in (Dylla & Wallgrün, 2007; Condotta et al., 2008). However, there is still a need for solutions that scale well to the large constraint networks occurring in potential application domains such as GIS.

3.5 Quantification

A symbolic qualitative scene description can be difficult to grasp for humans, particularly if many objects are involved. It is usually much easier to understand such a scene by looking at one particular drawing of it even though one has to be aware that this drawing is just one quantitative example among infinitely many ones subsumed by the qualitative description. Hence, often after doing some computation with qualitative relations one would want to visualize the result. This demand should be supported by QSTR toolboxes, at least by offering the service of computing one solution, i.e., quantitative example, for a given consistent scenario. Analogously to the qualification service, we call this process of computing a quantitative scene description from a qualitative scene description *quantification*. The process was already illustrated in Figure 4. Based on this service one could then visualize complete solution spaces by visualizing the individidual consistent scenarios comprised by a general constraint network. Unlike the quantification of cardinal direction relations as shown in Figure 4, which can be achieved by multiple sorting, quantification techniques for most qualitative calculi have not been studied yet. An universal quantification technique in the sense of composition-based reasoning applicable to all calculi would be most desirable. Also, good quantification methods would not compute some quantitative realization, but render a scene from which all qualitative relations can be easily read off (e.g., avoiding object distances visually too small) and support humans in understanding the image, for example by making use of preferred mental models (see Rauh et al., 2005) whenever possible.

3.6 Geometric Reasoning with Relation Semantics

Applications may require us to handle qualitative relations from different calculi or to mix qualitative and quantitative information. So far there are few approaches to handle relations beyond those captured in a single calculus. Unfortunately, the relation algebraic approach taken by the bipath-consistency method is often too weak (Wölfl & Westphal, 2009). Therefore, we see the need to research additional means to address these application-relevant topics. One promising approach to tackle these unsolved reasoning problems in QSTR can be found in the field of real algebraic geometry (see Basu, Pollack, & Roy, 2006). In order to employ methods from real algebraic geometry, QSTR constraints must be written as systems of multivariate polynomials and reasoning tasks need to be posed as tests for solvability of the system of equations. Algebraic geometry is a domain-level method that allows us to perform general geometric reasoning tasks. Thus, we are not limited to reasoning about what can be modeled using one specific calculus as we are in classical constraint-based qualitative reasoning. Using algebraic geometry we can also, for instance, determine composition tables from a specification of a set of relations, i.e., geometric reasoning allows us to instantiate a qualitative calculus. In a first study we have evaluated the effectiveness of such an approach to computing operation tables (composition, converse) for qualitative constraint calculi (Wolter & Moshagen, 2008). Since the motivation of qualitative approaches is to only make distinctions necessary in the context of a given task, qualitative representations are highly task-dependent. Therefore, one ideally would like to only name qualitative relations useful for a specific task and reasoning methods would be instantiated automatically. Although this goal remains subject to further research, geometric reasoning using methods of real algebraic geometry is a promising approach.

4 The SparQ Toolbox

With our own spatial reasoning toolbox SparQ[2] we aim at providing a tool collection that connects qualitative reasoning to applications. We provide the toolbox as free software licensed under the GNU GPL license; it is available from `http://www.sfbtr8.uni-bremen.de/project/r3/sparq/` and can be used in most POSIX-compliant operating systems. Our goal is to provide methods for all the different tasks we identified in the different application areas. Currently, we provide all standard constraint-based reasoning tools for binary and ternary calculi, means for connecting quantitative to qualitative data, tools for calculi analysis, and some manipulation of constraint-based knowledge bases—see Figure 6. Users can easily specify their own calculi or they can take advantage of the growing calculi repository that comes along with the toolbox. As heuristics are important to efficient reasoning, SparQ automatically computes reasoning heuristics for any calculus description provided—no manual work is required.

[2] **Spa**tial **R**easoning done **Q**ualtitatively

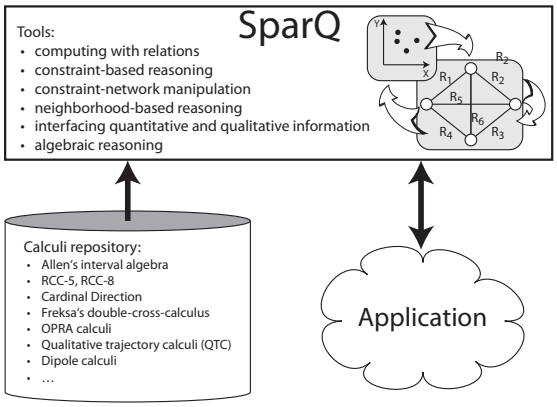


Figure 6: Methods provided by the SparQ toolbox

4.1 SparQ by Example

Let us consider a very simple example to illustrate how the different services provided by SparQ can be applied in the context of a GIS. The task in this example is to integrate two spatial knowledge bases of which one is a purely quantitative database and the other only provides qualitative information. After merging the two knowledge sources, the combined information shall be visualized for the user.

The two databases DB_1 and DB_2 are depicted in Figure 7 and Figure 8. They both contain information about the positions of buildings at the Bremen university campus. DB_1 contains positions within an Cartesian coordinate system extracted from a map. DB_2 contains qualitative direction information given in terms of relations from a qualitative spatial calculus. It could, for instance, be the result of a human writing down his knowledge about the buildings from memory. It also contains one object, the glass hall (GH), which is not contained in DB_1. The qualitative calculus we are using in this example is the cardinal direction calculus (Ligozat, 1998) which distinguishes nine base relations n, nw, w, sw, s, se, e, ne, and eq (for equal). Figure 9 shows the definition file of the cardinal direction calculus in SparQ. It specifies basic properties of the calculus such as its arity, base entities, base relations, and its identity relation and defines the converse and composition operations. Optionally, calculi definitions may contain an algebraic specification of the semantic of the base relations. Relations are defined by the zero sets of multivariate polynomials over the field of reals. SparQ offers a general method for interfacing qualitative and quantitative information using the algebraic specification. Identifying which relation holds between objects that are specified quantitatively (qualification), is performed by checking for each relation whether its algebraic specification is satisfied by the quantitative data. SparQ uses exact computations with rational numbers to avoid numerical problems. Of course, the algebraic approach is restricted to domains that can be formalized over a real-valued domain and whose relation

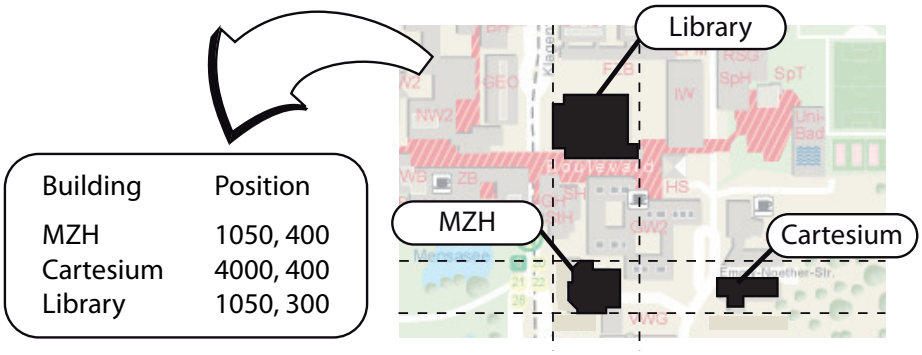


Figure 7: Quantitative information derived from a map is stored in DB_1

The MZH is **southwest, south** or **southeast** of the Library.
The Cartesium is **southeast** of the Library.
The GH is **northwest** of the MZH.
The GH is **southwest** of the Library.
The GH is **southwest** of the Cartesium.

Figure 8: Qualitative information stemming from a human is stored in DB_2

can be modeled algebraically. We note that topological calculi like RCC-8 are defined for abstract domains and cannot be specified this way. However, to our knowledge, all other qualitative calculi documented in the literature can be formalized using polynomial equations.

Before we start with merging the information from DB_1 and DB_2, it is convenient to assign the information from both databases to variables inside SparQ for later use. To do this, the GIS software would send the following commands to SparQ:

```
let DB1 = ((MZH 1050 400) (CARTESIUM 4000 400) (LIBRARY 1050 300))
<ENTER>
```

and

```
let DB2 = ((MZH (sw s se) LIBRARY) (CARTESIUM se LIBRARY)
  (GH nw MZH) (GH sw LIBRARY) (GH sw CARTESIUM))                <ENTER>
```

As we see, the quantitative scene description for DB_1 in SparQ is simple a list of tuples describing individual objects. The object tuples consist of an identifier (MZH, CARTESIUM, etc.) and the parameters to specify an object from the underlying domain (here points in the plane). The qualitative scene description for DB_2 is a list of relational tuples of the (*object* $\times$ (*list-of-base-relations*) $\times$ *object*) describing the relations holding between the objects which might be unions of base relations like in the case of the MZH and the LIBRARY.

```
(def-calculus "Cardinal direction calculus (cardir)"
  :arity          :binary
  :basis-entity   :2d-point
  :base-relations (N NE E SE S SW W NW EQ)
  :identity-relation EQ

  :converse-operation
    ((N S) ; A (N) B => B (S) A
     (NE  SW)
     (E W)
     (SE  NW)
     ... )

  :composition-operation
    ((N N N) ; A (N) B, B (N) C => A (N) C
     (N S (N EQ S))
     (N E NE)
     (N W NW)
     ... )

  :algebraic-specification
    ((EQ ((1 ((ax 1))) = (1 ((bx 1))))  ;  EQ :<=> 1*ax^1 = 1*bx^1,
         ((1 ((ay 1))) = (1 ((by 1))))) ;          1*ay^1 = 1*by^1
     (NW ((1 ((ax 1))) < (1 ((bx 1))))
         ((1 ((ay 1))) > (1 ((by 1)))))
     (W  ((1 ((ax 1))) < (1 ((bx 1))))
         ((1 ((ay 1))) = (1 ((by 1)))))
     ... )
```

Figure 9: Definition of cardinal direction calculus in SparQ

As a first step of merging the information from DB_1 and DB_2 we need to turn the quantitative information from DB_1 into qualitative information to make it comparable to the information in DB_2. Hence, our GIS software would send the following request to SparQ, asking it to perform a qualification using the cardinal direction calculus (abbreviated cardir in SparQ) and store the result for later use in the variable `DB1Qual`:

```
let DB1Qual = qualify cardir all $DB1                    <ENTER>
```

The parameter "all" has the effect that the resulting constraint network contains the relations holding between every pair of objects. Hence, the output of SparQ corresponding to the content now stored in `DB1Qual` is:

```
((MZH w CARTESIUM) (MZH s LIBRARY) (CARTESIUM se LIBRARY))
```

We now can merge the two constraint networks stored in `DB1Qual` and `DB2`. The intuitive way to do this is to merge the networks by taking the intersection of corresponding constraints. This can be done in SparQ using the refine command which is a subcommand of the constraint-reasoning module:

```
let DB3 = constraint-reasoning cardir refine $DB1Qual $DB2     <ENTER>
```

The result is:

```
((CARTESIUM se LIBRARY) (GH sw CARTESIUM) (GH sw LIBRARY)
  (GH nw MZH) (MZH w CARTESIUM) (MZH s LIBRARY))
```

However, intersecting both contraint networks is the strictest form of merging information and the result may very well be inconsistent if the networks contain conflicting information. Hence, our software could check whether the result of the merging is consistent or not. To do this, it uses the following SparQ command:

```
constraint-reasoning cardir check-consistency check $DB3       <ENTER>
```

with the following result:

```
Not consistent.
```

By using the subcommand `check-consistency` we request SparQ to decide consistency of the specified constraint network using the appropriate decision method for this calculus. In this specific case, the computationally cheap algebraic closure algorithm suffices to decide consistency. For other calculi like e.g., RCC-8, one would need to perform a backtracking search to decide existence of a refined network that is algebraically closed and only contains relations known to be tractable (see Renz, 2002). Knowing that enforcing algebraic closure is suitable in our case, we could also invoke this tool directly.

```
constraint-reasoning cardir algebraic-closure $DB3             <ENTER>
```

resulting in:

```
...
;; Checking GH->CARTESIUM:(sw) = GH->MZH:(nw) o MZH->CARTESIUM:(w)
refining to GH->CARTESIUM:()
...
Not consistent.
```

This time we also get some additional information telling us where the algebraic closure algorithm failed: composing GH nw MZH and MZH w CARTESIUM yields GH nw CARTESIUM which contradicts the given information GH se CARTESIUM.

As we do not want to continue with an inconsistent database, we would like to relax the constraints in DB_3 until we end up with a consistent network. More precisely, we want to find a network that is as similar as possible to DB_3 but is consistent. Relaxation in this sense is realized within the neighborhood-reasoning module of SparQ because, as we previously mentioned, similarity of constraint networks can be defined based on the notion of conceptual neighborhood. The call to find a minimally relaxed network that is consistent is:

```
let DB4 = neighborhood-reasoning cardir relax $DB3                <ENTER>
```

yielding

```
((CARTESIUM (se) LIBRARY ) (GH (sw) CARTESIUM) (GH (sw) LIBRARY)
  (GH (nw) MZH) (MZH (sw) CARTESIUM) (MZH (s) LIBRARY))
```

Comparing this DB_4 to DB_3, we see that only the relation between MZH and CARTESIUM has changed from west to southwest. The resulting network is indeed consistent:

```
constraint-reasoning cardir scenario-consistency check $DB4     <ENTER>
Consistent.
```

Finally, we want to visualize the content of our merged database DB_4. Since DB_4 is already a scenario, our GIS software can directly call

```
let DB4Quan = quantify cardir $DB4                                <ENTER>
```

which yields a quantitative scene description like

```
((MZH 20 0) (CARTESIUM 100 20) (LIBRARY 20 100) (GH 0 10))
```

also depicted in Figure 10. If DB_4 would have been a non-atomic network, we could have first used

```
constraint-reasoning cardir scenario-consistency all $DB4       <ENTER>
```

which yields a list of all consistent scenarios of DB_4 which then could be quantified individually.

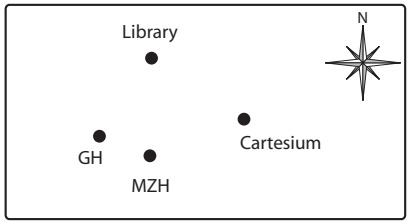


Figure 10: Resulting quantitative scene

4.2 SparQ Roadmap

Besides improving existing tools, we are working on long- and mid-term goals that require more research. We envisage the following three main directions:

algebraic geometry reasoning As discussed in Section 3.6, algebraic reasoning methods can provide alternative means to qualitative reasoning. This is particularly valuable in cases in which QSTR methods are not sufficient—for instance, when constraint-based reasoning using composition tables fails to decided consistency of constraint networks as, for example the case with relative position calculi (see,e.g., Lücke et al., 2008). Second, algebraic geometry is calculi-independent, so qualitative reasoning tasks can be handled that involve relations from different calculi. For tasks involving multiple calculi we would obtain more effective means than, for example, the bipath-consistency (Gerivini & Renz, 2002; Westphal & Wölfl, 2008) method which is currently the only possibility of combining calculi without manually instantiating and analyzing a new, combined calculus.

universal method for mapping qualitative to quantitative information So far, only some specific methods for individual calculi are known that can map qualitative to quantitative information; for many calculi no method is known. This stands in contrast to the constraint-based reasoning methods in QSTR which are purely syntactically (they only make use of operation tables). However, interfacing of quantitative and qualitative knowledge builds on the semantics of qualitative relations. Therefore, we enhance calculi with a semantic specification using algebraic equations (this part is already contained in SparQ). We are currently developing algebraic reasoning techniques that provide us with a universal method capable of computing a quantitative scene for any (consistent) qualitative constraint network. Additionally, we will consider cognitive principles to generate scenes that humans would preferably reconstruct (cp. Rauh et al., 2005), thereby easing the visual perception.

constraint-based querying We identified the need to being able to find parts in a constraint-network which are identical or similar to a given query

network. Technically speaking, the task of detecting occurrences of constraint networks within a larger network as we discussed earlier is a form of subgraph isomorphy which in its general form is known to be NP-hard. Therefore, it is important to employ effective heuristics to solve this task. We will research means to compute these heuristics automatically and to make use of calculus operations.

5 Conclusions and Outlook

We argued that one issue impeding the dissemination of QSTR techniques in applications is a lack of software solutions which allow for an easy integration of QSTR methods by application developers. Existing toolboxes and theoretical research so far have focused on constraint-based reasoning and the consistency problem. By looking at three exemplary application domains we showed that classical constraint-based reasoning is only a small part of the reasoning that is required in these domains. Tasks that need to be carried out include mapping of quantitative data to qualitative and vice versa, efficient means for representing and retrieving qualitative knowledge, and neighborhood-reasoning. Existing toolboxes need to be extended to accommodate these demands. We discussed how well these additional reasoning tasks are currently understood theoretically and to which extent they are currently supported by the toolboxes. We then focused on our own QSTR toolbox SparQ and described what is currently already feasible in SparQ as well as our roadmap for the future towards offering all the discussed reasoning services in an easy-to-use way.

We also identified some important research questions. Currently, qualitative representations are not integrated with quantitative data, in particular it remains an open question which general approaches allow qualitative knowledge to be externalized diagrammatically. In general, integrating qualitative and quantitative approaches is important to systems which are classically based on quantitative data such as GIS. Moreover, qualitative representation and reasoning has so far been studied using one specific calculus at a time. Reasoning methods remain confined to using only one qualitative calculus at a time—in order to combine calculi a new calculus needs to be developed. However, as qualitative representations are per se task-dependent, many applications need to build on several calculi at the same time, or to utilize self-defined relations. Thus, general calculi-independent reasoning methods need to be developed.

We believe that developing tools that accomodate the practial needs arising in applications is key to overcoming the lack of successful QSTR applications. Ultimately, we envision that QSTR toolboxes will offer a variety of means for representing and manipulating spatial and temporal knowledge which are accessible through some kind of spatial programming language—similar to algebra toolboxes that offer a variety of mathematical tools today.

Acknowledgments

We would like to thank Christian Freksa, Frank Dylla, Lutz Frommberger, Mehul Bhatt, Jae Hee Lee, Paolo Fogliaroni, and Giorgio de Felice for valuable discussions on the material presented here. This work was carried out in the framework of the SFB/TR 8 Spatial Cognition, project R3 [Q-Shape], and the IRTG GRK 1498 Semantic Integration of Geospatial Information. Financial support by the Deutsche Forschungsgemeinschaft is gratefully acknowledged.

References

Bar-Shalom, Y., & Fortmann, T. E. (1988). *Tracking and data association.* Academic Press.

Basu, S., Pollack, R., & Roy, M.-F. (2006). *Algorithms in real algebraic geometry* (2nd ed.). Springer.

Belussi, A., Catania, B., & Podestà, P. (2005). Towards topological consistency and similarity of multiresolution geographical maps. In *GIS '05: Proceedings of the 13th annual ACM international workshop on geographic information systems* (pp. 220–229). New York, NY, USA: ACM.

Bhatt, M., Dylla, F., & Hois, J. (2009). Spatio-terminological inference for the design of ambient environments. In K. S. Hornsby, C. Claramunt, M. Denis, & G. Ligozat (Eds.), *Spatial information theory, 9th international conference, COSIT 2009, aber wrac'h, france, september 21-25, 2009 proceedings* (Vol. 5756). Springer.

Bittner, T., Donnelly, M., & Smith, B. (2009). A spatio-temporal ontology for geographic information integration. *International Journal on Geographic Information Science*, *23*(6), 765–798.

Bojduj, B., Weber, B., Richter, K.-F., & Bertel, S. (2008). Computer aided architectural design: Wayfinding complexity analysis. In *Proceedings of the 12th international conference on computer supported collaborative work in design (CSCWD)* (p. 919-924).

Bruns, H. T., & Egenhofer, M. J. (1996). Similarity of spatial scenes. In *Seventh international symposium on spatial data handling* (p. 173-184). Taylor & Francis.

Burrough, P. A. (1986). *Principles of geographical information systems for land resources assessment.* Oxford University Press.

Castellanos, J., Neira, J., & Tardós, J. D. (2006). Map building and slam algorithms. In S. S. Ge & F. L. Lewis (Eds.), *Autonomous mobile robots: Sensing, control, decision-making and applications.* Taylor & Francis.

Cohn, A. G., Gotts, N. M., Randell, D. A., Cui, Z., Bennett, B., & Gooday, J. M. (1997). Exploiting temporal continuity in qualitative spatial calculi. In R. G. Golledge & M. J. Egenhofer (Eds.), *Spatial and temporal reasoning in geographical information systems.* Elsevier. Specialist Meeting on Time in Geographic Space, Lake Arrowhead, California (1993).

Cohn, A. G., & Hazarika, S. M. (2001). Qualitative spatial representation and reasoning: an overview. *Fundamenta Informaticae*, *46*, 1-29.

Cohn, A. G., & Renz, J. (2007). Qualitative spatial reasoning. In F. van Harmelen, V. Lifschitz, & B. Porter (Eds.), *Handbook of knowledge representation* (pp. 551–596). Elsevier.

Condotta, J.-F., Kaci, S., & Schwind, N. (2008). A framework for merging qualitative constraints networks. In D. Wilson & H. C. Lane (Eds.), *Proceedings of the twenty-first international Florida artificial intelligence research society conference, may 15-17, 2008, Coconut Grove, Florida, USA* (p. 586-591). AAAI Press.

Condotta, J.-F., Ligozat, G., & Saade, M. (2006). A generic toolkit for n-ary qualitative temporal and spatial calculi. In *Proceedings of the 13th international symposium on temporal representation and reasoning (TIME'06)* (pp. 78 – 86). Budapest, Hungary.

Dechter, R., Meiri, I., & Pearl, J. (1991). Temporal constraint networks. *Artificial Intelligence*, *49*(1-3), 61–95.

Duckham, M., & Kulik, L. (2005). A formal model of obfuscation and negotiation for location privacy. In H.-W. Gellersen, R. Want, & A. Schmidt (Eds.), *Pervasive* (Vol. 3468, p. 152-170). Springer.

Duckham, M., & Worboys, M. F. (2005). An algebraic approach to automated geospatial information fusion. *International Journal of Geographical Information Science*, *19*(5), 537-557.

Dudek, G., Freedman, P., & Hadjres, S. (1996). Using multiple models for environmental mapping. *Journal of Robotic Systems*, *13*(8), 539-559.

Dylla, F., & Wallgrün, J. O. (2007). Qualitative spatial reasoning with conceptual neighborhoods for agent control. *Journal of Intelligent and Robotic Systems*, *48*(1), 55–78.

Egenhofer, M. J. (1989). A formal definition of binary topological relationships. In *3rd international conference on foundations of data organization and algorithms* (pp. 457–472). Springer.

Egenhofer, M. J. (1994). Spatial SQL: A query and presentation language. *IEEE Transactions on Knowledge and Data Engineering*, *6*(1), 86–95.

Egenhofer, M. J., Clementini, E., & Felice, P. D. (1994). Evaluating inconsistencies among multiple representations. In *Sixth international symposium on spatial data handling* (pp. 901–920).

Egenhofer, M. J., & Mark, D. M. (1995). Naive geography. In *Proceedings of COSIT'95* (Vol. 988, p. 1-15). Springer.

Fonseca, F. T., Egenhofer, M. J., Agouris, P., & Câmara, G. (2002). Using ontologies for integrated geographic information systems. *Transactions on GIS*, *6*(3), 231-257.

Fox, D., Burgard, W., & Thrun, S. (1999). Markov localization for mobile robots in dynamic environments. *Journal of Artificial Intelligence Research*, *11*, 391–427.

Frank, A. (1991). Qualitative spatial reasoning about cardinal directions. In *Proceedings of the american congress on surveying and mapping (ACSM-ASPRS)* (pp. 148–167). Baltimore, Maryland, USA.

Freksa, C. (1991). Conceptual neighborhood and its role in temporal and spatial reasoning. In M. Singh & L. Travé-Massuyès (Eds.), *Decision support systems and qualitative reasoning* (pp. 181 – 187). North-Holland, Amsterdam.

Freksa, C. (1992). Temporal reasoning based on semi-intervals. *Artificial Intelligence*, *54*(1), 199–227.

Galton, A. (2000). *Qualitative spatial change*. Oxford University Press.

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.*

Gerivini, A., & Renz, J. (2002). Combining topological and size information for spatial reasoning. *Artificial Intelligence*, *137*(1–2), 1–42.

Grimson, W. E. L. (1990). *Object recognition by computer: The role of geometric constraints*. Cambridge (MA), USA: MIT Press.

Hähnel, D., Burgard, W., Wegbreit, B., & Thrun, S. (2003). Towards lazy data association in SLAM. In *Proceedings of the 11th international symposium of robotics research (ISRR'03).*

Hearnshaw, H. M., & Unwin, D. J. (1994). *Visualization in geographical information systems*. Chichester: John Wiley & Sons.

Janowicz, K., Raubal, M., Schwering, A., & Kuhn, W. (2008). Semantic similarity measurement and geospatial applications. *Transactions in GIS*, *12*(6), 651-659.

Krieg-Brückner, B., & Shi, H. (2006). Orientation calculi and route graphs: Towards semantic representations for route descriptions. In *Geographic information science - fourth international conference, GIScience* (Vol. 4197, pp. 234–250). Springer.

Kuipers, B. (1994). *Qualitative reasoning: Modeling and simulation with incomplete knowledge*. Cambridge, MA, USA: MIT Press.

Kuipers, B. (2000). The spatial semantic hierarchy. *Artificial Intelligence*, *119*, 191–233.

Kuipers, B., Modayil, J., Beeson, P., MacMahon, M., & Savelli, F. (2004, April). Local metrical and global topological maps in the hybrid spatial semantic hierarchy. In *Proceedings of the IEEE international conference on robotics and automation (ICRA-04).* New Orleans, LA, USA.

Leonard, J. J., & Durrant-Whyte, H. F. (1991). Simultaneous map building and localization for an autonomous mobile robot. In *Proceedings of IEEE/RSJ international workshop on intelligent robots and systems* (p. 1442-1447).

Levitt, T. S., & Lawton, D. T. (1990). Qualitative navigation for mobile robots. *Artificial Intelligence*, *44*, 305-360.

Ligozat, G. (1998). Reasoning about cardinal directions. *Journal of Visual Languages and Computing*, *9*, 23-44.

Ligozat, G., & Renz, J. (2004). What is a qualitative calculus? A general framework. In C. Zhang, H. W. Guesgen, & W. K. Yeap (Eds.), *PRICAI 2004: Trends in artificial intelligence — 8th pacific rim international conference on artificial intelligence* (pp. 53–64). Springer.

Lovett, A. M., Dehghani, M., & Forbus, K. D. (2007). Incremental learning of

perceptual categories for open-domain sketch recognition. In M. M. Veloso (Ed.), *IJCAI* (p. 447-452).

Lücke, D., Mossakowski, T., & Wolter, D. (2008). Qualitative reasoning about convex relations. In C. Freksa, S. N. Newcombe, P. Gärdenfors, & S. Wölfl (Eds.), *Spatial Cognition VI* (Vol. 5248, pp. 426–440). Springer.

Moratz, R. (2009). Ambiguous landmark problems in cognitive robotics: A benchmark for qualitative position calculi. In B. Nebel & S. Wölfl (Eds.), *Papers from the 2009 aaai spring symposium benchmarking of qualitative spatial and temporal reasoning systems* (pp. 17–22). AAAI Press; Menlo Park, CA.

Moratz, R., Nebel, B., & Freksa, C. (2003). Qualitative spatial reasoning about relative position: The tradeoff between strong formal properties and successful reasoning about route graphs. In C. Freksa, W. Brauer, C. Habel, & K. F. Wender (Eds.), *Spatial cognition III* (Vol. 2685, pp. 385–400). Springer.

Moratz, R., & Tenbrink, T. (2006). Spatial reference in linguistic human-robot interaction: Iterative, empirically supported development of a model of projective relations. *Spatial Cognition and Computation*, *6*(1), 63–106.

Moratz, R., & Wallgrün, J. O. (2003). Spatial reasoning about relative orientation and distance for robot exploration. In W. Kuhn, M. F. Worboys, & S. Timpf (Eds.), *Spatial information theory: Foundations of geographic information science. conference on spatial information theory (COSIT)* (p. 61-74). Springer.

Neira, J., & Tardós, J. D. (2001). Data association in stochastic mapping using the joint compability test. *IEEE Transactions on Robotics and Automation*, *17*, 890-897.

OpenGIS Consortium. (1999). *OpenGIS simple features implementation specification for SQL rev. 1.1.* OpenGIS Project Document 99-049.

Ragni, M., & Wölfl, S. (2005, September). Temporalizing spatial calculi—on generalized neighborhood graphs. In *Proceedings of the 28th German conference on Artificial Intelligence (KI 2005).* Koblenz, Germany.

Ragni, M., & Wölfl, S. (2006). Temporalizing cardinal directions: From constraint satisfaction to planning. In P. Doherty, J. Mylopoulos, & C. A. Welty (Eds.), *KR* (p. 472-480). AAAI Press.

Randell, D. A., Cui, Z., & Cohn, A. (1992). A spatial logic based on regions and connection. In B. Nebel, C. Rich, & W. Swartout (Eds.), *Principles of knowledge representation and reasoning: Proceedings of the third international conference (KR'92)* (pp. 165–176). San Mateo, CA: Morgan Kaufmann.

Rauh, R., Hagen, C., Knauff, M., Kuß, T., Schlieder, C., & Strube, G. (2005). From preferred to alternative mental models in spatial reasoning. *Spatial Cognition and Computation*, *5*, 239-269.

Remolina, E., & Kuipers, B. (2004). Towards a general theory of topological maps. *Artificial Intelligence*, *152*(1), 47-104.

Renz, J. (2002). *Qualitative spatial reasoning with topological information* (Vol. LNCS 2293). Springer.

Renz, J. (2007). Qualitative spatial and temporal reasoning: Efficient algorithms for everyone. In *Proceedings of the 20th international joint conference on artificial intelligence (IJCAI-07)* (p. 526-531). Hyderabad, India.

Renz, J., & Nebel, B. (2007). Qualitative spatial reasoning using constraint calculi. In M. Aiello, I. E. Pratt-Hartmann, & J. F. van Benthem (Eds.), *Handbook of spatial logics* (pp. 161–215). Springer.

Richter, K.-F., Weber, B., Bojduj, B., & Bertel, S. (to appear). Supporting the designer's and the user's perspectives in computer-aided architectural design. *Advanced Engineering Informatics*. Available from `http://dx.doi.org/10.1016/j.aei.2009.08.012`

Rodríguez, M. A., & Egenhofer, M. J. (2004). Comparing geospatial entity classes: An asymmetric and context-dependent similarity measure. *International Journal of Geographical Information Science*, *18*(3), 229-256.

Schlieder, C. (1993). Representing visible locations for qualitative navigation. In N. Piera-Carrete & M. Singh (Eds.), *Qualitative reasoning and decision technologies* (pp. 523–532).

Schultz, C. P. L., Amor, R., Lobb, B., & Guesgen, H. W. (2009). Qualitative design support for engineering and architecture. *Advanced Engineering Informatics*, *23*(1), 68–80.

Schwering, A. (2008). Approaches to semantic similarity measurement for geospatial data: A survey. *Transactions in GIS*, *12*(1), 5–29.

Smith, M. J. de, Goodchild, M. F., & Longley, P. A. (2007). *Geospatial analysis: A comprehensive guide to principles, techniques and software tools.* Troubador Publishing.

Steinhauer, H. J. (2008). Object configuration reconstruction from incomplete binary object relation descriptions. In A. R. Dengel, K. Berns, & T. M. Breul (Eds.), *KI 2008: Advances in artificial intelligence* (p. 348-355). Springer.

Thrun, S. (2002). *Robotic mapping: A survey* (Tech. Rep.). Carnegie Mellon University, Computer Science Department.

Wallgrün, J. O. (2009). Exploiting qualitative spatial constraints for multi-hypothesis topological map learning. In K. S. Hornsby, C. Claramunt, M. Denis, & G. Ligozat (Eds.), *Spatial information theory, 9th international conference, COSIT 2009, aber wrac'h, france, september 21-25, 2009 proceedings* (Vol. 5756, p. 141-158). Springer.

Wallgrün, J. O. (2010). *Hierarchical Voronoi graphs – Spatial representation and reasoning for mobile robots.* Springer.

Wallgrün, J. O., Frommberger, L., Wolter, D., Dylla, F., & Freksa, C. (2007). Qualitative spatial representation and reasoning in the SparQ-toolbox. In *Spatial cognition V: Reasoning, action, interaction: International conference spatial cognition 2006.* Bremen, Germany.

Westphal, M., & Wölfl, S. (2008). Bipath consistency revisited. In M. Ghallab, C. Spyropoulos, N. Fakotakis, & N. Avouris (Eds.), *Proceedings of the 18th european conference on artificial intelligence (ECAI 2008).* Amsterdam: IOS Press.

Westphal, M., & Wölfl, S. (2009). Qualitative CSP, finite CSP, and SAT: Com-

paring methods for qualitative constraint-based reasoning. In *Proceedings of IJCAI-09.*

Winter, S. (1998). Bridging vector and raster representation in GIS. In *GIS '98: Proceedings of the 6th acm international symposium on advances in geographic information systems* (pp. 57–62). New York, NY, USA: ACM.

Wölfl, S., & Westphal, M. (2009). On combinations of binary qualitative constraint calculi. In *Proceedings of IJCAI-09.*

Wolter, D. (2008). *Spatial representation and reasoning for robot mapping—a shape-based approach* (Vol. 48). Springer.

Wolter, D., Freksa, C., & Latecki, L. J. (2008). Towards a generalization of self-localization. In M. E. Jefferies & W. K. Yeap (Eds.), *Robot and cognitive approaches to spatial mapping.* Springer.

Wolter, D., & Moshagen, L. (2008). Algebraic methods for analyzing qualitative spatio-temporal calculi. In *Proceedings of ECAI-2008 workshop on spatial and temporal reasoning.*

Glossary

SparQ a qualitative spatial reasoning toolbox
`http://www.sfbtr8.uni-bremen.de/project/r3/sparq/`

qualitative relation relation (in the mathematical sense) between objects in a spatio-temporal domain that represents a meaningful category, e.g., "north of"

qualitative calculus relation algebraic structure comprising qualitative relations and operations

algebraic closure decision method for deciding consistency in qualitative constraint problems

conceptual neighborhood qualitative relations are conceptually neighbored if the categories they represent are connected, e.g., "smaller" is neighbored with "same size", but not with "larger"

constraint-based reasoning symbolic reasoning (e.g., deduction) with qualitative relations using the constraint-based semantics

neighborhood-based reasoning symbolic reasoning (e.g., deduction) with qualitative relations using conceptual neighborhoods