Relation Variables in Qualitative Spatial Reasoning

Sebastian Brand

National Research Institute for Mathematics and Computer Science (CWI)
P.O. Box 94079, 1090 GB, Amsterdam, The Netherlands
Sebastian.Brand@cwi.nl

Abstract. We study an alternative to the prevailing approach to modelling qualitative spatial reasoning (QSR) problems as constraint satisfaction problems. In the standard approach, a relation between objects is a constraint whereas in the alternative approach it is a variable.
By being declarative, the relation-variable approach greatly simplifies integration and implementation of QSR. To substantiate this point, we discuss several specific QSR algorithms from the literature which in the relation-variable approach reduce to the customary constraint propagation algorithm enforcing generalised arc-consistency.

1 Introduction

Qualitative spatial representation and reasoning (QSR) [6] lends itself well to modelling by constraints. In the standard approach, a spatial object, such as a region, is described by a variable, and the qualitative relation between spatial objects, such as a topological relation between two regions, contributes a constraint. For many QSR calculi, it is known that if all the constraints represent definite (base) relations and path-consistency (PC) holds, then this description of a spatial scene is consistent. If the relation is not fully specified, the corresponding constraint is a disjunction of basic constraints. By establishing PC, such a disjunctive constraint is refined in view of the constraints with which it shares a variable. A combination of PC with search over the disjunctive constraints decides the consistency of indefinite scene descriptions.

We examine here an alternative constraint-based formulation of QSR. In this approach, a spatial object is a constant, and the relation between spatial objects is a variable. We call this the *relation-variable* approach, in contrast to the conventional *relation-constraint* approach above. Although modelling QSR with relation variables is not original, see [27], it is mentioned very rarely. This fact surprises in view of the advantages of this approach. In particular, the following two important issues are tackled successfully:

Integration. Space has several aspects that can be characterised qualitatively, such as size, shape, orientation. These aspects are interdependent, but no convenient canonical representation exists that provides a link (the role of time points in temporal reasoning). Spatial reasoning problems in practice are also not likely to occur in pure form. They may be embedded into a non-spatial context, or contain application-specific side constraints.

The relation-variable approach to QSR is declarative in a strict sense and is thus well-suited for these integration problems.

Systems. Typical current constraint solving platforms focus on domain reduction, and accordingly provide convenient access to variable domains. Modifying the constraint network, on the other hand, is usually difficult. This task is, however, required for enforcing PC.

A formulation of QSR according to the relation-variable approach means that generic domain-reducing propagation algorithms and conventional constraint solving platforms can be used instead of dedicated spatial reasoning systems.

Plan of the paper. We begin by introducing briefly the necessary constraint solving concepts and methods, and qualitative spatial reasoning, using the example of the RCC-8 calculus. The next section presents in-depth the two modelling approaches for constraint-based QSR. In the following sections, we discuss several aspects of space and contrast the relation-variable and relation-domain approach. We finally mention some new modelling options, and end with a summary.

1.1 Constraint Satisfaction

Recent coverage of the field can be found in [1, 8, 12].

Consider a sequence $X = x_1, \ldots, x_m$ of pairwise different variables with respective domains $D_1, \ldots, D_m$. By a *constraint* C on X, written $C(X)$, we mean a subset of $D_1 \times \cdots \times D_m$. The arity of C is m. A *constraint satisfaction problem (CSP)* consists of a finite sequence of variables $X = x_1, \ldots, x_n$ with respective domains $\mathcal{D} = D_1, \ldots, D_n$, and a finite set $\mathcal{C}$ of constraints, each on a subsequence of X. We write it as $\langle \mathcal{C}; x_1 \in D_1, \ldots, x_n \in D_n \rangle$, or shorter as $\langle \mathcal{C}; X \in \mathcal{D} \rangle$. Given an element $d = d_1, \ldots, d_n$ of $D_1 \times \ldots \times D_n$ and a subsequence $Y = x_{i_1}, \ldots, x_{i_\ell}$ of X we denote by $d[Y]$ the sequence $d_{i_1}, \ldots, d_{i_\ell}$; in particular, we have $d[x_k] = d_k$. A *solution* to $\langle \mathcal{C}; X \in \mathcal{D} \rangle$ is an element $d \in \mathcal{D}$ such that for each constraint $C \in \mathcal{C}$ on the variables Y we have $d[Y] \in C$.

Constraint propagation. One method to establish satisfiability of CSPs when the search space is finite is systematic search for a solution. For reducing the search space and overall search effort, constraint propagation is often very useful; the principle is to replace a given CSP by another one that is equivalent with respect to the solutions but that is easier to solve. Constraint propagation is typically characterised by the resulting *local consistency*. The two notions most relevant for this paper are:

***Path Consistency (PC)*:** A CSP of binary constraints is path-consistent [24] if for every triple of variables x, y, z

$$C(x,z) = \{\, (a,c) \mid b \text{ exists s.t. } (a,b) \in C(x,y) \text{ and } (b,c) \in C(y,z) \,\}\,.$$

It is assumed here that a unique constraint $C(u,w)$ for each pair of variables u, w exists, and that $C(u,w) = C^{-1}(w,u)$.

Generalised Arc-Consistency (GAC): A constraint $C(X)$ is generalised arc-consistent [23] if for all $x_k \in X$ and all $a \in D_k$

$$d \in C(X) \quad \text{exists such that} \quad d[x_k] = a.$$

In short, every domain value must participate in a local solution.
A CSP is generalised arc-consistent if each of its constraints is.

For example, the CSP $\langle x + y = z;\ x, y, z \in \{1, 2, 3\}\rangle$ can be reduced to $\langle x + y = z;\ x, y \in \{1, 2\}, z \in \{2, 3\}\rangle$ which is GAC.

Enforcing PC means reducing constraints but not domains, whereas enforcing GAC means reducing domains but not constraints.

A number of generic methods to establish GAC for a constraint are known, and many constraint solving systems have implementations. One example is the *GAC-schema* [3] available in ILOG Solver [16].

1.2 Qualitative Spatial Reasoning

The topological calculus RCC-8 [25] is one of the best-known formalisations in spatial reasoning. We use it to illustrate a number of concepts. In RCC-8 one distinguishes 8 topological relations between two regions, see Fig. 1: *disconnected, externally connected, partially overlapping, equal, tangential proper part, non-tangential proper part*, and inverses of the latter two. These are denoted $\mathsf{DC}, \mathsf{EC}, \mathsf{PO}, \mathsf{EQ}, \mathsf{TPP}, \mathsf{NTPP}, \mathsf{TPPi}, \mathsf{NTPPi}$, respectively; together they form a set that we call $\mathsf{RCC8}$.

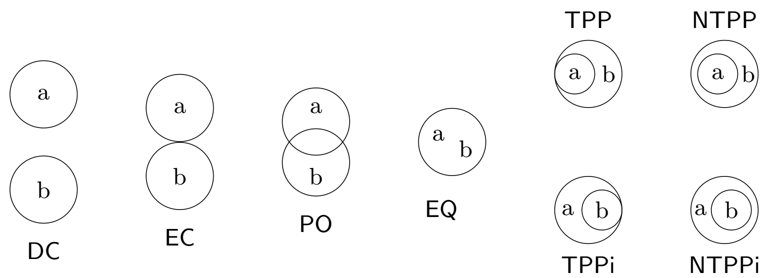

Fig. 1. RCC-8 relations (2D example)

Jointly exhaustive and pairwise disjoint. Any two spatial regions are in one and exactly one of the RCC-8 relations to each other.

Composition table. Considering the triple $R_{a,b}, R_{b,c}, R_{a,c}$ of relations between regions a, b, c, one finds that not all triples of RCC-8 relations are semantically feasible. The consistent triples are collected in the RCC-8 composition table. It contains 193 relation triples, such as ($\mathsf{NTPP}, \mathsf{EC}, \mathsf{DC}$). Bennett [2] proved that compositional consistency entails global consistency: if for all triples of regions the relations between them respect the composition table then this topological scenario is consistent.

Converse relation table. In analogy to the composition table, it is helpful to think of a converse relation table consisting of the 8 pairs (R, Ri) of RCC-8 relations such that Ri is the converse of R. It contains for example (EQ, EQ) and ($\mathsf{TPP}, \mathsf{TPPi}$). If we agree on ($\mathsf{EQ}$) for the relation of a region with itself then the converse relation table follows from the composition table.

2 Approaches to Constraint-based QSR

A spatial topological scenario consists of a set of region names denoted by *Regions*, and possibly some restrictions on the topological relation for regions pairs. A scenario is fully specified if for each region pair exactly one RCC-8 relation is given.

We examine now how scenarios can be modelled as constraint satisfaction problems. We continue using topology with RCC-8 as an example, but most of the concepts below are immediately transferable to other spatial aspects.

2.1 Relations as Constraints

In this conventional approach, *Regions* is considered to be a set of region variables. Their infinite domain is the set of all spatial regions in the underlying topological space; for example, if we model 2D space then a region variable represents a set of points in the plane. Information about the topological relation between two regions is expressed as a binary constraint *Rel* that corresponds to a subset of $\mathsf{RCC8}$. One usually writes this in infix notation as

$$\text{constraint} \quad x \mathit{Rel}\, y \qquad \text{where} \quad \mathit{Rel} \subseteq \mathsf{RCC8} \quad \text{and} \quad x, y \in \mathit{Regions}.$$

Such a CSP describes a possibly partially specified scenario. Whether a corresponding fully specified and satisfiable scenario exists is checked by path-consistency and search over the relations. A PC-enforcing algorithm revises the constraints between regions according to the converse relation and composition tables of RCC-8, and search branches over disjunctive constraints.

Establishing satisfiability of a scenario processes only the constraints, for compositional consistency. The variables remain unassigned.

2.2 Relations as Variables

Here we interpret every element of *Regions* as a constant. The topological relation between two regions is a variable with a subset of $\mathsf{RCC8}$ as its domain. Such a relation variable exists for each ordered pair of regions, and we collect all these variables in an array *Rel*. We write an individual relation as

$$\text{variable} \quad \mathit{Rel}[a, b] \qquad \text{where} \quad \mathit{Rel}[a, b] \subseteq \mathsf{RCC8} \quad \text{and} \quad a, b \in \mathit{Regions}.$$

Integrity constraints. Relation converse and composition in this setting are captured at the constraint level. The binary constraint conv represents the converse relation table:

$$\mathsf{conv}(\ Rel[a,b],\ Rel[b,a]\ ) \qquad \text{for all} \quad \{a,b\} \subseteq Regions.$$

The composition table is represented by the ternary constraint comp, with

$$\mathsf{comp}(\ Rel[a,b],\ Rel[b,c],\ Rel[a,c]\ ) \qquad \text{for all} \quad \{a,b,c\} \subseteq Regions.$$

In presence of

$$Rel[a,a] = \mathsf{EQ} \qquad \text{for all} \quad a \in Regions$$

and a conv constraint for all pairs of different regions, one comp constraint per three different regions suffices.

2.3 Comments

By modelling the items of interest as variables and static information as constraints, the relation-variable approach yields plain finite-domain CSPs in which the solutions (i. e., assignments) are relevant. There is a straightforward correspondence between a solution and a fully specified, consistent scenario. Obtaining the latter from a partially specified scenario amounts to the standard task of solving a finite-domain CSP.

Constructing a relation-variable model means finding integrity constraints that embody the intended semantics. Once that has been established, the origin or meaning of the constraints is irrelevant. For example, a constraint solver can ignore whether comp represents the composition operation in a relation algebra; we also discuss examples below in which other restrictions on the relations must be satisfied. There is thus a clear distinction between specification and execution. The relation-variable approach is declarative in a strict sense.

Constraint propagation. The relation-variable approach is independent of the particular constraint solving method. We could, however, choose a solver based on search and propagation, and furthermore we could choose a GAC-enforcing propagation algorithm.

Path consistency in the relation-constraint approach and generalised arc-consistency in the relation-variable approach simulate each other. This can be seen by analysing, in both approaches, the removal of one topological relation from the disjunctive constraint $a\ Rel\ b$, or from the domain of the variable $Rel[a,b]$, respectively. The reason in both cases must be the lack of supporting relations between a, c and b, c, for some third region c; that is, compositional consistency.

Complexity. It is perhaps not surprising but useful to mention that establishing the respective local consistency in either approach (i. e., PC and GAC) requires the same computational effort. Let n denote the number of regions. Enforcing PC by an algorithm as the one given in [21] requires time in $O(n^3)$ [22]. For this, one assumes that one PC step, restricting $a\ Rel\ c$ by $a\ Rel\ b$ and $b\ Rel\ c$, takes constant time.

Analogue reasoning entails that GAC can be enforced in constant time on a single $\mathsf{comp}(Rel[a, b], Rel[b, c], Rel[a, c])$ constraint — observe that the three variables have domains of size at most eight. In this way, the overall time complexity depends only on the number of such constraints, and is thus in $O(n^3)$.

Previous work. Tsang [27] describes the relation-variable approach in qualitative temporal reasoning, a field similar to QSR. The idea appears not to have caught on, however. One reason is probably that integration in temporal reasoning is simpler because the canonical representation of time points on the real line exists. By referring to its end points, a time interval can directly be related to its duration or another time interval. Space, in contrast, has no such convenient canonical representation — but many aspects to be integrated.

In QSR, the possibility of the relation-variable approach is mentioned occasionally in passing, but without examining its potential. For actually modelling and solving QSR problems using relation variables I am only aware of [1, pages 30-33], which deals with a single aspect (topology) only.

3 Relation Variables in Use

An essential advantage of the relation-variable approach is that the relevant information is available in variables. This means that linking pieces of information reduces to merely stating additional constraints on the variables. In that way, embedding a QSR problem into an application context or adding side restrictions, for example, can be dealt with easily and declaratively.

We illustrate the issue of composite models with the case of aspect integration.

3.1 Combining Topology and Size

Following Gerevini and Renz [13], we study scenarios combining topological and size information. We collect information about both these aspects and their link in one CSP.

Let n be the number of regions.

Topological aspect. As in Section 2.2, the

$n \times n$ array $TopoRel$

of RCC-8 relation variables stores the topological relation between two regions. The integrity constraints $\mathsf{conv}_{\mathrm{RCC8}}$, $\mathsf{comp}_{\mathrm{RCC8}}$ need to hold.

Size aspect. Relative size of regions is captured by one of $\{<,=,>\}$, as in [13]. The

$$n \times n \text{ array } \mathit{SizeRel}$$

of variables stores the relative sizes of region pairs. The converse relation and composition tables are straightforward; the integrity constraints are

$$\begin{array}{rll} \mathsf{conv}_{\mathrm{Size}} = \{\, (<,>),\ (=,=),\ (>,<) \,\}, & & \text{and} \\ \mathsf{comp}_{\mathrm{Size}} = \{\, (<,<,<),\ (<,=,<),\ \ldots\} & & \text{(13 triples).} \end{array}$$

Linking the aspects. The topological relation between two regions is dependent on their relative size. A table with this information is given in [13], it contains rules such as the following:

$$\begin{array}{lll} \mathit{TopoRel}[x,y] = \mathsf{TPP} & \text{implies} & \mathit{SizeRel}[x,y] = (<), \\ \mathit{SizeRel}[x,y] = (=) & \text{implies} & \mathit{TopoRel}[x,y] \in \{\mathsf{DC}, \mathsf{EC}, \mathsf{PO}, \mathsf{EQ}\}. \end{array}$$

In [13], these rules represent a meta constraint. Here, we infer the linking constraint

$$\mathsf{link}_{\mathrm{Topo\&Size}} = \{\, (\mathsf{TPP},<),\ (\mathsf{DC},=),\ \ldots\} \qquad \text{(14 pairs)}$$

which is to be stated as

$$\mathsf{link}_{\mathrm{Topo\&Size}}(\ \mathit{TopoRel}[a,b],\ \mathit{SizeRel}[a,b]\ )$$

for all regions a, b.

Example. Let us pick up the combined scenario from [13, p. 14]. Five regions, denoted by $\{0,\ldots,4\}$, are constrained by

$$\begin{array}{ll} \mathit{TopoRel}[0,2] \in \{\mathsf{TPP}, \mathsf{EQ}\} & \mathit{SizeRel}[0,2] \in \{<\} \\ \mathit{TopoRel}[1,0] \in \{\mathsf{TPP}, \mathsf{EQ}, \mathsf{PO}\} & \mathit{SizeRel}[3,1] \in \{<,=\} \\ \mathit{TopoRel}[1,2] \in \{\mathsf{TPP}, \mathsf{EQ}\} & \mathit{SizeRel}[2,4] \in \{<,=\} \\ \mathit{TopoRel}[4,3] \in \{\mathsf{TPP}, \mathsf{EQ}\} & \end{array}$$

Independently, the topological and the size scenarios are consistent while the combined scenario is not. It is pointed out in [13] that naive propagation scheduling schemes do not suffice to detect inconsistency.

A formulation of this scenario as a combined topological & size CSP in the relation-variable approach is straightforward. The resulting CSP can be entered into a constraint programming platform such as ECL^{i}PS^e [28]. ECL^{i}PS^e is focused on search and domain-reducing propagation; in particular, it offers a GAC-enforcing propagation algorithm for user-defined constraints. Given our CSP in ECL^{i}PS^e, solely executing GAC-propagation for all constraints yields failure, which proves that this CSP is inconsistent. □

For the same purpose but within the relation-constraint approach, Gerevini and Renz proposed a new algorithm called BIPATH-CONSISTENCY [13]. Its principle is the computation of path-consistency for both types of relations in an interleaved fashion while taking into account the interdependency. The $\mathsf{link}_{\mathrm{Topo\&Size}}$ constraint is in essence treated as a *meta constraint* on the algorithm level. Moreover, the BIPATH-CONSISTENCY algorithm fixes in part the order of propagation.

The relation-variable method, on the other hand, is declarative; all information is in the five types of constraints. They are handled by repeated, interleaved calls to the same GAC-enforcing algorithm. The actual propagation order is irrelevant for the result.

BIPATH-CONSISTENCY is restricted to combining two types of relations (e. g., two aspects of space). In contrast, the relation-variable approach is compositional in the sense that adding a third aspect, such as morphology [7] or orientation, is straightforward. It amounts to formulating integrity constraints (e. g., $\mathsf{conv}, \mathsf{comp}$), linking constraints to each of the already present aspects, and a constraint linking all three aspects. Some of these constraints may be logically redundant.

3.2 Combining Cardinal Directions and Topology

In orientation, another important aspect of space, one studies the relation of two objects, the primary and the reference object, with respect to a frame of reference. It is thus inherently a ternary relation, but by agreeing on the frame of reference, a binary relation is obtained.

The binary relation approach is realised in the cardinal direction model [9], based on the geographic (compass) directions. Points as well as regions have been studied as the objects to be oriented. The point-based models can be cast in the relation-variable approach analogously to topology, Section 2.2. For instance, Frank [9] distinguishes the jointly exhaustive and pairwise disjoint relations $\mathsf{N}, \mathsf{NW}, \mathsf{W}, \ldots$ for points; denoting North, Northwest, West, and so on. Ligozat [20] gives a composition table.

Orienting regions. Goyal and Egenhofer [15] and Skiadopoulos and Koubarakis [26] study a more expressive model, in which the oriented objects are regions. The exact shape of the primary region is taken into account, and a ninth atomic relation B exists, describing overlap of the primary region and the axes-parallel minimum bounding box of the reference region. *Sets* of the atomic relations are then used to describe directional information. In this way, for example, the position of South America for an observer located in Ecuador can be fully described by the set $\{\mathsf{B}, \mathsf{N}, \mathsf{NE}, \mathsf{E}, \mathsf{SE}, \mathsf{S}\}$. In contrast, the position of Ecuador with respect to South America is just $\{\mathsf{B}\}$.

Relation variables for directional information are thus naturally *set variables*: they take their value from a set of sets of constants, unlike relation variables for topology and size whose domain is a set of atomic constants.

For each pair a, b of regions, the direction is a relation variable

$$\mathit{DirRel}[a,b] \in \mathcal{P}(\mathsf{Dir}) \qquad \text{where} \quad \mathsf{Dir} = \{\mathsf{B}, \mathsf{N}, \mathsf{NW}, \ldots, \mathsf{NE}\}.$$

$\mathcal{P}$ denotes the power set function.

Integrity constraints. A restriction on the set values that $\mathit{DirRel}[a,b]$ can take arises if a, b are internally connected regions, which is often assumed. Only 218 of the 512 subsets of Dir are then semantically possible. This knowledge can be represented in a unary integrity constraint, which for example allows $\{\mathsf{N}, \mathsf{NE}, \mathsf{E}\}$ but excludes $\{\mathsf{N}, \mathsf{S}\}$. The usual integrity constraints comp and conv can be derived from studies of composition [26] and converse [5] (but it is outside of our focus whether these are the only integrity constraints needed).

Integration with topology. Let us briefly consider linking directional information to topology. The relevant knowledge could be expressed by rules as

$$\begin{array}{lll} \mathit{TopoRel}[x,y] \in \{\mathsf{EQ}, \mathsf{NTPP}, \mathsf{TPP}\} & \text{implies} & \mathit{DirRel}[x,y] = \{\mathsf{B}\}, \\ \mathit{TopoRel}[x,y] \in \{\mathsf{NTPPi}, \mathsf{TPPi}\} & \text{implies} & \mathit{DirRel}[x,y] \supseteq \{\mathsf{B}\}, \end{array}$$

from which a constraint $\mathsf{link}_{\text{Topo\&Dir}}$ can be defined. It is to be stated as

$$\mathsf{link}_{\text{Topo\&Dir}}(\ \mathit{TopoRel}[a,b],\ \mathit{DirRel}[a,b]\ )$$

for all regions a, b. We now have some components of a combined cardinal directions & topology model. It can be given to any sufficiently expressive constraint solver, which in particular would provide constraints on set variables.

Constraint solving with set variables is discussed in [14]. Many contemporary constraint programming systems support set variables.

3.3 Cyclic Ordering of Orientations with Relation Variables

From the several formalisations of orientation information with an explicit frame of reference, let us examine the approach of Isli and Cohn to cyclic ordering of 2D orientations [19]. Here, the spatial objects are orientations, i. e. directed lines. At the root of the framework is the qualitative classification of the angle $\alpha = \sphericalangle(a, b)$ between the two orientations a and b by

$$\mathsf{Or}(\alpha) = \begin{cases} \mathsf{e} \quad \text{(equal)} & \text{if} \quad \alpha = 0, \\ \mathsf{l} \quad \text{(left)} & \text{if} \quad 0 < \alpha < \pi, \\ \mathsf{o} \quad \text{(opposite)} & \text{if} \quad \alpha = \pi, \\ \mathsf{r} \quad \text{(right)} & \text{if} \quad \pi < \alpha < 2\pi \end{cases}$$

into the jointly exhaustive and pairwise disjoint relations $\mathsf{e}, \mathsf{l}, \mathsf{o}, \mathsf{r}$. See Fig. 2 for an illustration. For three orientations a, b, c, we now consider the triple

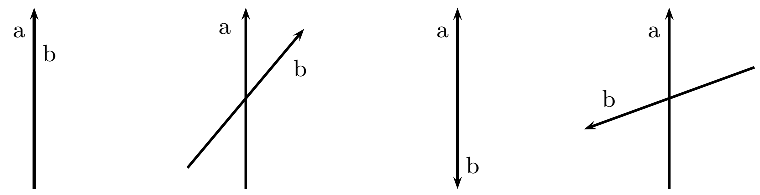

Fig. 2. The relations $\mathsf{e}, \mathsf{l}, \mathsf{o}, \mathsf{r}$ of a pair of orientations

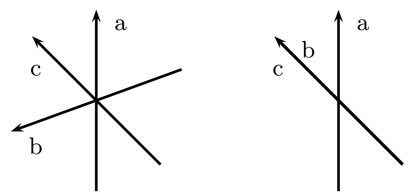

Fig. 3. The Cyc relations lrl, lel, and rol of a triple of orientations

$$\langle\ \mathsf{Or}(\lhd(b,a)),\ \mathsf{Or}(\lhd(c,b)),\ \mathsf{Or}(\lhd(c,a))\ \rangle.$$

Of all 4^3 triples over $\{\mathsf{e}, \mathsf{l}, \mathsf{o}, \mathsf{r}\}$, only 24 combinations are geometrically possible. We denote this set by Cyc. Fig. 3 shows three of its elements.

Such cyclic ordering information can be expressed within the relation variable approach in an array *CycRel*, which in particular is ternary. We have thus a relation variable

$$CycRel[a,b,c] \in \mathsf{Cyc} \qquad \text{with} \quad \mathsf{Cyc} = \{\mathsf{lrl}, \mathsf{orl}, \ldots, \mathsf{rle}\}$$

for every three orientations a, b, c. The integrity constraints here are

$$\mathsf{conv}(\ CycRel[a,b,c],\ CycRel[a,c,b]\ ),$$
$$\mathsf{comp}(\ CycRel[a,b,c],\ CycRel[a,c,d],\ CycRel[a,b,d]\ ),$$

and a new constraint

$$\mathsf{rotate}(\ CycRel[a,b,c],\ CycRel[c,a,b]\ ).$$

Details and definitions can be found in [19].

Working within the relation constraint approach, Isli and Cohn construct a new algorithm called *s4c* that enforces 4-consistency [11] on the ternary relation constraints that correspond to *CycRel*. They are able to prove that this algorithm decides consistency, i. e., 2D geometric feasibility, of fully specified scenarios. The *s4c* algorithm uses exactly the information that we represent in the conv, comp and $\mathsf{rotation}$ constraints. Consequently, we can conclude that in our relation variable model these constraints guarantee geometric consistency.

We hypothesise further that *s4c* in the relation constraint model propagates at most as much information as a GAC-enforcing algorithm does in our relation variable model. Intuitively, this should be clear: every possible reduction of a disjunctive constraint in the relation constraint model corresponds to a domain reduction of a relation variable in our model.

3.4 Combining Cardinal Direction with Relative Orientation

Isli [17, 18] studies the problem of exchanging information between a cardinal direction model for pairs of points as in Section 3.2, and a relative orientation model for triples of points, derived from Freksa and Zimmermann's formalisation [10]. This problem is again similar to combining topology and size, Section 3.1. Isli works with the relation-constraints and proposes a new algorithm for this integration issue.

We formulate a relation-variable model. The cardinal direction subproblem can straightforwardly be expressed in this approach; we omit the obvious details here. The relative orientation subproblem leads to a model similar to that of orientations in the preceding section; in particular, it is based on a ternary array. The arrays in the combined model are:

$$n \times n \times n \text{ array } \mathit{ROrientRel}, \text{ and}$$
$$n \times n \text{ array } \mathit{CDirRel},$$

if we assume n points.

For linking the two models, Isli [18] devises functions for both directions of the information transfer. They can be transformed into the two constraints

$$\mathsf{link}_{\mathrm{CD}\to\mathrm{RO}}(\,\mathit{CDirRel}[a,b],\ \mathit{CDirRel}[b,c],\ \mathit{ROrientRel}[a,b,c]\,),$$
$$\mathsf{link}_{\mathrm{CD}\leftarrow\mathrm{RO}}(\,\mathit{ROrientRel}[a,b,c],\ \mathit{CDirRel}[a,b],\ \mathit{CDirRel}[b,c],\ \mathit{CDirRel}[a,c]\,).$$

For the relation-constraint model it is necessary to treat the information in $\mathsf{link}_{\mathrm{CD}\to\mathrm{RO}}, \mathsf{link}_{\mathrm{CD}\leftarrow\mathrm{RO}}$ as meta-constraints, embedded inside an algorithm that moreover integrates *s4c* of [19] and a path-consistency algorithm.

Using relation variables, it suffices to state the constraints and provide a generic GAC-enforcing algorithm. Also, for a given triple of points, the first constraint $\mathsf{link}_{\mathrm{CD}\to\mathrm{RO}}$ should just be the restriction of the second constraint $\mathsf{link}_{\mathrm{CD}\leftarrow\mathrm{RO}}$ in which the variable $\mathit{CDirRel}[a,c]$ is projected away. The former constraint is then redundant, and we just need one constraint

$$\mathsf{link}_{\mathrm{CD\&RO}}(\,\mathit{ROrientRel}[a,b,c],\ \mathit{CDirRel}[a,b],\ \mathit{CDirRel}[b,c],\ \mathit{CDirRel}[a,c]\,).$$

On the grounds that both the relation-variable and the relation-constraint approach are based on the same semantic information, for one embedded in an algorithm, for the other in constraints, we conclude that both accept exactly the same point configuration scenarios.

4 Extensions

Variables ranging over spatial objects. In the relation-variable model, spatial objects are denoted by constants. An *object variable*, whose domain is the set of object constants, has thus a different meaning than in the relation-constraint approach. This issue is best demonstrated by an example. Suppose we wish to identify two regions among all given regions such that

- the first is smaller than the second, and
- they are disconnected or externally connected.

We use topological and size information, formalised as in Section 3.1, so we have arrays *SizeRel* and *TopoRel* recording the qualitative relations. Let *Regions* be the set of the n region constants. We define the

region variables x_1, x_2

whose domain is the set *Regions*, and constrain them by

$$\mathit{SizeRel}[x_1, x_2] = (<), \tag{C_1}$$

$$\mathit{TopoRel}[x_1, x_2] \in \{\mathsf{DC}, \mathsf{EC}\}. \tag{C_2}$$

C_1 is a constraint on the variables x_1, x_2 and on all size relation variables in the array *SizeRel*. Namely, region constants $r_1, r_2 \in \mathit{Regions}$ must be assigned to x_1, x_2 such that the size relation variable $\mathit{SizeRel}[r_1, r_2]$ is assigned a '$<$'.

We call such constraints, in which arrays are indexed by variables instead of constants, *array constraints*. They are a generalisation of the better-known element constraint, which corresponds to a one-dimensional array indexed by a variable. Constraint propagation to establish GAC for array constraints is studied in [4]. The constraint programming system ILOG Solver [16] accepts and propagates array constraints.

Reasoning about spatial change. It is not difficult to augment a relation-variable model with temporal information. It suffices to add a new time index to each array of qualitative relations, and to link the new time-annotated scenarios appropriately. We extend *Rel* from a binary to a ternary array such that

$\mathit{Rel}[a, b, t]$

is a variable specifying the relation between the spatial objects a and b at time t. Suppose we view time as linear and discrete, such that only atomic relational changes can occur between subsequent time points. We can specify these atomic changes (the so-called conceptual neighbourhood) by pairs of qualitative relations and define accordingly a new binary constraint neighbour. For example, the pair $(\mathsf{DC}, \mathsf{EC})$ in the constraint $\mathsf{neighbour}_{\mathrm{Topo}}$ indicates that the topological relation *disconnected* between two regions may change in one time step to *externally connected*. The neighbour constraint is then stated on all variable pairs $(\mathit{Rel}[a, b, t], \mathit{Rel}[a, b, t'])$ where t directly precedes t' temporally.

5 Summary

We have presented an alternative formulation of qualitative spatial reasoning problems as constraint satisfaction problems. Contrary to the conventional approach, we model qualitative relations as variables. Uncertain relational information is naturally expressed by variables with domains; consistency of this information is naturally expressed by static constraints. The propagation of these constraints is a well-understood issue in research on constraint programming, and corresponding generic algorithms are provided by many constraint solving systems.

While the principle of the relation-variable approach is not new, the advantages of applying it to QSR, especially for integration tasks, have so far very rarely been realised. We have argued that several algorithms that are custom-designed for integrating spatial aspects become unnecessary if a relation-variable model and a generic GAC-establishing constraint propagation algorithm is used: the Bipath-consistency algorithm of [13], the *s4c* algorithm of [19], the algorithm combining *s4c* and a path-consistency algorithm of [18]. We have shown how the relation-variable approach can accommodate composite qualitative relations as investigated in [5, 26] with the help of set variables and constraints. We have indicated that extending or combining a relation-variable model often consists mainly in defining appropriate constraints, contrary to what is the case in the relation-constraint approach where new algorithms must be designed.

Finally, we remark that the strictly declarative model that results from using relation-variables can be solved by any sufficiently expressive solver of CSPs. This includes typical CP systems based on search and propagation, but also for example solvers based on local search.

References

1. K. R. Apt. *Principles of Constraint Programming.* Cambridge University Press, 2003.
2. B. Bennett. Determining consistency of topological relations. *Constraints*, 2:213–225, 1998.
3. C. Bessière and J.-C. Régin. Arc consistency for general constraint networks: preliminary results. In *Proc. of 15th International Joint Conference on Artificial Intelligence (IJCAI'97)*, pages 398–404, 1997.
4. S. Brand. Constraint propagation in presence of arrays. In K. R. Apt, R. Barták, E. Monfroy, and F. Rossi, editors, *Proc. of 6th Workshop of the ERCIM Working Group on Constraints*, 2001.
5. S. Cicerone and P. Di Felice. Cardinal directions between spatial objects: the pairwise-consistency problem. *Information Sciences*, 164:165–188, 2004.
6. A. G. Cohn and S. M. Hazarika. Qualitative spatial representation and reasoning: An overview. *Fundamenta Informaticae*, 46(1-2):1–29, 2001.
7. M. Cristani. The complexity of reasoning about spatial congruence. *Journal of Artificial Intelligence Research*, 11:361–390, 1999.
8. R. Dechter. *Constraint Processing.* Morgan Kaufmann, 2003.

9. A. U. Frank. Qualitative spatial reasoning about distance and directions in geographic space. *Journal of Visual Languages and Computing*, 3:343–373, 1992.
10. C. Freksa and K. Zimmermann. On the utilization of spatial structures for cognitively plausible and efficient reasoning. In *Proc. of IEEE International Conference on Systems, Man, and Cybernetics*, pages 18–21. IEEE, 1992.
11. E. C. Freuder. Synthesizing constraint expressions. *Communications of the ACM*, 21(11):958–966, 1978.
12. T. Frühwirth and S. Abdennadher. *Essentials of Constraint Programming.* Springer, 2003.
13. A. Gerevini and J. Renz. Combining topological and size constraints for spatial reasoning. *Artificial Intelligence*, 137(1-2):1–42, 2002.
14. C. Gervet. Interval propagation to reason about sets: Definition and implementation of a practical language. *Constraints*, 1(3):191–244, 1997.
15. R. K. Goyal and M. J. Egenhofer. The direction-relation matrix: A representation of direction relations for extended spatial objects. In *Proc. of UCGIS Annual Assembly and Summer Retreat*, 1997.
16. ILOG S.A. *Solver 5.1 Reference Manual*, 2001.
17. A. Isli. Combining cardinal direction relations and relative orientation relations in qualitative spatial reasoning. Technical report, University of Hamburg, Dept. of Informatics, 2003.
18. A. Isli. Combining cardinal direction relations and other orientation relations in QSR. In *Proc. of 8th International Symposium on Artificial Intelligence and Mathematics (AI&M'04)*, 2004.
19. A. Isli and A. G. Cohn. A new approach to cyclic ordering of 2D orientations using ternary relation algebras. *Artificial Intelligence*, 122(1-2):137–187, 2000.
20. G. Ligozat. Reasoning about cardinal directions. *Journal of Visual Languages and Computing*, 9(1):23–44, 1998.
21. A. K. Mackworth. Consistency in networks of relations. *Artificial Intelligence*, 8(1):118–126, 1977.
22. A. K. Mackworth and E. C. Freuder. The complexity of some polynomial network algorithms for constraint satisfaction problems. *Artificial Intelligence*, 25:65–74, 1985.
23. R. Mohr and G. Masini. Good old discrete relaxation. In Y. Kodratoff, editor, *Proc. of European Conference on Artificial Intelligence (ECAI'88)*, pages 651–656. Pitman publishers, 1988.
24. U. Montanari. Networks of constraints: Fundamental properties and applications to picture processing. *Information Science*, 7:95–132, 1974.
25. D. A. Randell, Z. Cui, and A. G. Cohn. A spatial logic based on regions and connection. In B. Nebel, C. Rich, and W. R. Swartout, editors, *Proc. of 2nd International Conference on Principles of Knowledge Representation and Reasoning (KR'92)*, pages 165–176. Morgan Kaufmann, 1992.
26. S. Skiadopoulos and M. Koubarakis. Composing cardinal direction relations. In C.S. Jensen, M. Schneider, B. Seeger, and V.J. Tsotras, editors, *Proc. of 7th International Symposium on Advances in Spatial and Temporal Databases (SSTD'01)*, volume 2121 of *LNCS*, pages 371–386. Springer, 2001.
27. E. P. K. Tsang. The consistent labeling problem in temporal reasoning. In K. S. H. Forbus, editor, *Proc. of 6th National Conference on Artificial Intelligence (AAAI'87)*, pages 251–255. AAAI Press, 1987.
28. M. G. Wallace, S. Novello, and J. Schimpf. ECLiPSe: A platform for constraint logic programming. *ICL Systems Journal*, 12(1):159–200, 1997.