StarVars—Effective Reasoning about Relative Directions*

Jae Hee Lee
University of Bremen
Germany
jay@sfbtr8.uni-bremen.de

Jochen Renz
The Australian National University
Australia
jochen.renz@anu.edu.au

Diedrich Wolter
University of Bremen
Germany
dwolter@sfbtr8.uni-bremen.de

Abstract

Relative direction information is very commonly used. Observers typically describe their environment by specifying the relative directions in which they see other objects or other people from their point of view. Or they receive navigation instructions with respect to their point of view, for example, turn left at the next intersection. However, it is surprisingly hard to integrate relative direction information obtained from different observers, and to reconstruct a model of the environment or the locations of the observers based on this information. Despite intensive research, there is currently no algorithm that can effectively integrate this information: this problem is NP-hard, but not known to be in NP, even if we only use left and right relations.

In this paper we present a novel qualitative representation, StarVars, that can solve these problems. It is an extension of the STAR calculus [Renz and Mitra, 2004]) by a VARiable interpretation of the orientation of observers. We show that reasoning in StarVars is in NP and present the first algorithm that allows us to effectively integrate relative direction information from different observers.

1 Introduction

Research in qualitative spatial reasoning (QSR) develops the theoretical foundations for abstract, symbolical representation and reasoning with spatial knowledge. Most spatial representations developed in QSR are so-called *qualitative calculi* [Ligozat and Renz, 2004b], which are spatial logics in the sense of [Aiello *et al.*, 2007] with the syntax of a constraint language, i.e., qualitative representations are conjuncts of constraint relations. A fundamental reasoning task for such representations is that of deciding consistency, i.e., to compute a yes/no answer to the question whether a spatial model exists that satisfies all constraint relations.

From the perspective of applications in autonomous agents, representations that can handle directional knowledge are particularly important. Directional knowledge describes the position of an object relative to the position and orientation of an observer, i.e., it uses an ego-centric frame of reference which allows observations of an agent to be described. This is essential for many navigation problems, including interpreting route directions and representation and reasoning with navigation rules. In the last 20 years, various approaches to representing directional knowledge have been proposed, starting with the works [Freksa, 1992; Ligozat, 1993] to, more recently, [Moratz, 2006; Mossakowski and Moratz, 2012]. It was however discovered that deciding consistency for these representations is NP-hard and it remains an open question whether an NP algorithm exists [Wolter and Lee, 2010]. Moreover, previous research has focused on deciding consistency of a qualitative representation, the question of how a spatial model can be computed remains largely unanswered.

This poses a severe limitation for human machine interaction, since it is not possible to visualize the represented spatial knowledge.

In this paper we develop the new constraint language *StarVars*, which allows us to represent relative direction knowledge of arbitrary granularity and accuracy.

We present an NP algorithm that computes a spatial model for a given StarVars constraint formula and fails if and only if the constraint formula is unsatisfiable. Doing so, we obtain an important answer to the two longstanding questions of how one can reason effectively with relative direction relations and how to present this knowledge visually.

2 Qualitative Spatial Reasoning

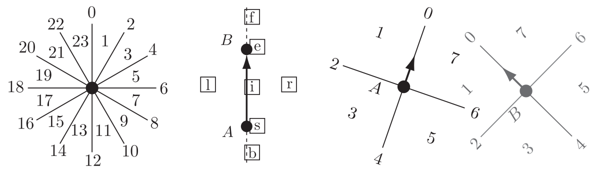
In QSR, constraint languages are used to represent spatial knowledge [Cohn and Renz, 2008]. Let a constraint language $\mathcal{L}$ defined as a finite collection $\mathcal{R}$ of binary relations on an infinite domain of spatial entities D. A central problem of QSR is solving the constraint satisfaction problem for $\mathcal{L}$ (CSP($\mathcal{L}$) for short), i.e., deciding the satisfiability of a formula

$$\bigwedge_{i,j\in\{1,\ldots,n\},i\neq j} v_i\ R_{ij}\ v_j, \tag{1}$$

where $v_1,\ldots,v_n$ are variables defined on D, and each R_{ij} is a finite union of members of $\mathcal{R}$.[1] We call R_{ij} a relation of $\mathcal{L}$ and formula (1) a CSP($\mathcal{L}$) instance. If R_{ij} is the

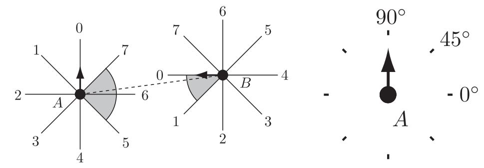
*Financial support by DFG (SFB/TR 8 Spatial Cognition, project R3-[Q-Shape]) is acknowledged.

[1] CSP($\mathcal{L}$) for ternary or n-ary relations is defined analogously.



(a) (b) (c)

Figure 1: (a) $\mathcal{STAR}_4$ relations. (b) The nine directional $\mathcal{LR}$ relations. (c) Relation $A\angle_7^1 B$ in $\mathcal{OPRA}_2$.



(a) (b)

Figure 2: (a) An $\mathcal{SV}_8$ object A with $\theta_A = 90°$. (b) Configuration of two $\mathcal{SV}_8$ objects, where $A[\![5..7]\!]B$ and $B[\![0]\!]A$ hold

union of all members of $\mathcal{R}$, we call it the *universal relation* of $\mathcal{L}$. If R_{ij} involves only one member of $\mathcal{R}$, then R_{ij} is said to be an *atomic* relation of $\mathcal{L}$, and we call a CSP($\mathcal{L}$) instance atomic, if all of its relations R_{ij} are atomic. Typically, in view of composition-based constraint propagation as reasoning method [Renz and Nebel, 2007], the members of $\mathcal{R}$ are supposed to be jointly exhaustive and pairwise disjoint (JEPD)[2] and form a partition scheme [Ligozat and Renz, 2004a]; a constraint language with this property is called a *qualitative calculus*.

2.1 Cardinal Direction Relations

One example of a spatial constraint language is $\mathcal{STAR}$ that represents qualitative directions between points in the plane [Renz and Mitra, 2004]. For each point p, $\mathcal{STAR}$ defines a number of direction sectors and half-lines that determine the spatial relationship of other points q with respect to p (see Figure 1a). The qualitative direction constraint pRq is the sector (or the half-line) R in which q is located relative to p. $\mathcal{STAR}_m$ allows us to adjust the granularity by varying the number m of lines that determine the direction sectors. The angular range of the sectors may be chosen freely, but commonly all sectors are equally large.

Given a choice of sectors, all direction information between any two points is expressed with respect to these sectors and half-lines whose orientation is globally aligned. Therefore, $\mathcal{STAR}$ relations are cardinal direction relations that do not depend on the orientation of the points.

2.2 Relative Direction Relations

Relative direction relations describe directions between two objects relative to a reference object. The most elementary system of directional relations is called FlipFlop or $\mathcal{LR}$ [Ligozat, 1993], see Figure 1b. The qualitative relations of $\mathcal{LR}$ are defined based on a reference system generated by a directed line connecting two points. The position of a third point is then categorized as to be either left or right of the line, or on 5 different segments of the reference line (two additional relations describe degenerate cases and are omitted here). Other point-based approaches like, for example, the direction relations "double cross" [Freksa, 1992] or TPCC [Moratz and Ragni, 2007] are refinements of $\mathcal{LR}$ relations.

Binary relations are sufficient to determine relative directions for objects comprising an intrinsic direction (e.g., the island is on the *starboard side* of a sailing ship). Domains considered in QSR are, for example, directed lines or oriented points, i.e., points that have been equipped with an orientation. One particularly interesting representative of this class is the set of $\mathcal{OPRA}_m$ relations [Moratz, 2006], which is based on the domain $\mathbb{R}^2 \times [0, 2\pi)$ of oriented points. Half-lines and angular sectors are instantiated to describe the position of one oriented point as seen from another. The relations of $\mathcal{OPRA}_m$ are defined with respect to a granularity parameter that determines how many sectors are used ($\mathcal{OPRA}_m$ uses m lines to divide the full circle evenly, giving $2m$ angular sectors and $2m$ half-lines). Figure 1c presents an example of an $\mathcal{OPRA}_2$ relation $\angle_7^1$. In the example, B is located in sector 7 as seen from A, which, in turn, is located in sector 1 as seen from B. $\mathcal{OPRA}_m$ is very similar to $\mathcal{STAR}$, but the sectors in $\mathcal{OPRA}_m$ depend on the relative orientation of the point, while for $\mathcal{STAR}$ their direction is globally fixed.

While representing knowledge in an ego-centric frame of reference is well-suited for applications with navigating agents, reasoning with relative direcions is not tractable. Even for the most elementary set of $\mathcal{LR}$'s directional relations "left of" and "right of" the problem of deciding whether an atomic instance of CSP($\mathcal{LR}$) is consistent is NP-hard [Wolter and Lee, 2010], while NP-membership has not been shown so far. Since all known point-based relative direction calculi are more expressive than $\mathcal{LR}$ they are all NP-hard and their NP-membership remains unclear as well. From the perspective of a practitioner wanting to utilize relative directional knowledge it may be even more impeding that no effective decision procedure is known so far. In particular, the composition-based reasoning widely used in QSR is ineffective for deciding consistency of CSPs involving directional relations [Lücke *et al.*, 2008].

3 The Spatial Constraint Language $\mathcal{SV}_m$

We will now define StarVars$_m$ or $\mathcal{SV}_m$—a constraint language for relative direction relations that integrates the expressiveness of relative direction representation with the well-behaved properties of cardinal directions. The relations of $\mathcal{SV}_m$ are obtained by generalizing $\mathcal{STAR}_m$ relations to accommodate for relative direction information. Our idea is to retain the fixed arrangement of sectors given by $\mathcal{STAR}_m$ relations but to interpret directions with respect to an observer's orientation. An example of two $\mathcal{SV}_8$ objects are shown in Figure 2a.

In detail, the constraint language $\mathcal{SV}_m$ is defined over the domain $\mathbb{R}^2 \times \Theta_m$, where $m \in \mathbb{N}, m \geq 2$ is the granularity

[2] A set of relations is JEPD, if for each pair of elements $(x, y) \in D \times D$, there is one and only one relation of $\mathcal{R}$ that contains the pair.

parameter and $\Theta_m = \{k \cdot \frac{360^\circ}{m} | k = 0, 1, \ldots, m-1\}$ the orientation domain. Given an oriented point $A = (x_A, y_A, \theta_A) \in \mathbb{R}^2 \times \Theta_m$, the value of $(x_A, y_A) \in \mathbb{R}^2$ determines the position and the value of $\theta_A \in \Theta_m$ the angle of the orientation of A. In Figure 2b an oriented point A in $\mathcal{SV}_8$ is illustrated with $\theta_A = 90^\circ$. By definition the higher the value m, the finer adjustable is the orientation of an $\mathcal{SV}_m$ object.

We define relations in $\mathcal{SV}_m$ as follows. Given a pair of oriented points, the position of the second point is described with respect to the first point which serves as reference. To this end, we partition the plane into m evenly sized angular sectors centered at the reference point—see Figure 2a for illustration. The sectors of the reference point are bounded by $m-1$ half-lines numbered $0, 1, \ldots, m-1$ counterclockwise with half-line 0 aligned with the orientation of the reference point. Sector s is bounded by half-lines s and $s+1 \pmod m$, where half-line s belongs to that sector and half-line $s+1 \pmod m$ does not. Oriented points A and B are said to be in relation $[\![s]\!]$ if B is positioned in sector s of A. Additionally, we establish a special relation $\mathbf{S}$ for the case of super-position, i.e. for two oriented points sharing the same position in $\mathbb{R}^2$ but not necessarily the same orientation. In summary, relations $[\![0]\!], [\![1]\!], \ldots, [\![m-1]\!], \mathbf{S}$ constitute the JEPD set $\mathcal{R}$ of atomic relations in $\mathcal{SV}_m$.

We will use the notation $[\![c..d]\!]$ as an abbreviation of relation $[\![c]\!] \cup [\![c+1]\!] \cup \cdots \cup [\![d-1]\!]$ (i.e., the angular sector bounded by half-lines c and d). Here, and throughout the paper all operations on numbers associated with $\mathcal{SV}_m$ relations are taken modulo m.

Example 1. A CSP($\mathcal{SV}_8$) instance $v_1 [\![6]\!] v_2 \wedge v_2 [\![0]\!] v_1$ is satisfiable as v_1 and v_2 can be instantiated with A and B as shown in Fig. 2a.

Example 2. A CSP($\mathcal{SV}_8$) instance $\bigwedge_{i,j \in \{1,2,3\}, i \neq j} v_i [\![1]\!] v_j$ cannot be satisfiable, because all three interior angles of the triangle which is spanned by v_1, v_2, v_3 are constrained to be less than 45° by the atomic relation $[\![1]\!]$. This contradicts the fact that the interior angles of a triangle add up to 180°.

4 NP-Hardness of CSP($\mathcal{SV}_2$)

In this section we show that deciding CSP($\mathcal{SV}_2$) is NP-hard. The NP-hardness result suggests that developing an algorithm parametrized with m that decides CSP($\mathcal{SV}_m$) in polynomial-time for all m is not possible.

For our proof we use a reduction from the BETWEENNESS problem, originally introduced as total ordering problem [Opatrny, 1979]. In BETWEENNESS we are given a set of constraints in the form $B(q_i, q_j, q_k)$, $i, j, k \in \{1, \ldots, n\}$ where q_i, q_j, q_k are variables that range over the set of rational numbers $\mathbb{Q}$. A constraint $B(q_i, q_j, q_k)$ stands for $(q_i < q_j < q_k) \vee (q_i > q_j > q_k)$, stating that "$q_j$ is between q_i and q_k". Deciding whether there exists a valuation satisfying all BETWEENNESS constraints is NP-complete [Opatrny, 1979].

Theorem 3. *CSP($\mathcal{SV}_2$) is NP-hard.*

Proof sketch. Given an instance χ of BETWEENNESS over n variables. We translate the BETWEENNESS constraints to CSP($\mathcal{SV}_2$) constraints. First, for every variable q_j occurring in a BETWEENNESS constraint we introduce a variable v_j^1 in $\mathcal{SV}_2$. If a variable q_j occurs t_j times as the second argument of a BETWEENNESS constraint we additionally introduce one new variable $v_j^2, v_j^3, \ldots, v_j^{t_j}$ in $\mathcal{SV}_2$ for every occurrence. We tie together all $v_j^1, v_j^2, \ldots, v_j^{t_j}$ by introducing constraints $v_j^\ell \mathbf{S} v_j^{\ell+1}$ for all $1 \leq \ell \leq t_j - 1$. Thus, we ensure that the points are co-located but possibly equipped with different orientations from $\Theta_2 = \{0^\circ, 180^\circ\}$. For every constraint $B(q_i, q_j, q_k)$ in χ we introduce two $\mathcal{SV}_2$ constraints, $v_i^1 [\![1]\!] v_j^\ell \wedge v_j^\ell [\![1]\!] v_k^1$, with ℓ being the ℓth appearance of q_j as second argument. All other variables are related via the universal relation to each other. Finally, we obtain a CSP($\mathcal{SV}_2$) instance ϕ involving only atomic relations and the universal relation, where the BETWEENNESS to CSP($\mathcal{SV}_2$) reduction is done in polynomial time in the number of variables.

We need to show that ϕ is satisfiable if and only if χ has a solution. First, assume χ is satisfiable. Then there exists a valuation function f that assigns to each variable q_j a point in $\mathbb{Q}$, such that all BETWEENNES constraints are satisfied. Then, by assigning $(0, f(q_j), \theta_j^\ell) \in \mathbb{R}^2 \times \Theta_2$ to each v_j^ℓ we obtain a valuation that satisfies each constraint $v_i^1 [\![1]\!] v_j^\ell \wedge v_j^\ell [\![1]\!] v_k^1$, where we choose $\theta_j^\ell = 0^\circ$ if $f(q_i) < f(q_j) < f(q_k)$ and $\theta_j^\ell = 180^\circ$ if $f(q_k) < f(q_j) < f(q_i)$.

Let us now consider ϕ is satisfiable. Then there is a valuation function that assigns to each variable v_j^ℓ a value $(x_j, y_j, \theta_j^\ell) \in \mathbb{R}^2 \times \Theta_2$, such that $v_i^1 [\![1]\!] v_j^\ell \wedge v_j^\ell [\![1]\!] v_k^1$ is satisfied. Wlog. we can assume that all x_j are zero, and all y_j rational, because a solution for ϕ is invariant under projection to the y-axis and a sufficiently small perturbation of y_j. Then by assigning to each q_j a rational number y_j for $j = 1, \ldots, n$ we obtain a solution for χ $\square$

5 The NP-Membership of CSP($\mathcal{SV}_m$)

In this section we give a decision procedure for $\mathcal{SV}_m$. This NP-membership proof establishes a connection to the problem of solving a system of linear inequalities which we later utilize to develop the practical decision procedure.

The following lemma from vector algebra establishes a connection between the determinant of two vectors in $\mathbb{R}^2$ and their relative directions. Recall that the determinant of a 2-by-2 matrix is defined as

$$\det \begin{pmatrix} a & b \\ c & d \end{pmatrix} = ad - bc. \tag{2}$$

Lemma 4. *Let two vectors $\vec{v}, \vec{w} \in \mathbb{R}^2$ be given and let $(\vec{v}, \vec{w})$ be a 2×2 matrix having $\vec{v}$ and $\vec{w}$ as its column vectors. Then $\vec{w}$ is to the left (right) of $\vec{v}$, if and only if the determinant of two vectors, i.e,* $\det(\vec{v}, \vec{w})$*, is greater (less) than 0. Vector $\vec{w}$ is parallel to $\vec{v}$, if and only if* $\det(\vec{v}, \vec{w}) = 0$.

Convex $\mathcal{SV}_m$ relations play an important role in the algorithm presented in the next section. We say that an $\mathcal{SV}_m$ relation $[\![c..d]\!]$ is *convex*, if for a given $\mathcal{SV}_m$ object A the sector of A that relation $[\![c..d]\!]$ describes is convex (i.e., the central angle of the sector is less or equals 180°). This is the case if and only if $\text{mod}(d-c,\, m) \leq m/2$. Any atomic relation is for example convex. In $\mathcal{SV}_8$ (cf. Figure 2a) relation $[\![5..1]\!]$ is convex. By contrast, relation $[\![5..2]\!]$ is not convex, because the union of sectors 5 to 2 in counterclockwise order span 225°.

Lemma 5. *A constraint $v_1[\![c..d]\!]v_2$ with a convex relation $[\![c..d]\!]$ can be formulated as a conjunction of nonlinear inequalities.*

Proof. Let $\vec{v} = (x_2, y_2) - (x_1, y_1) \in \mathbb{R}^2$ be a vector with initial point (x_1, y_1) and terminal point (x_2, y_2), where (x_1, y_1) and (x_2, y_2) are the positions of variables v_1 and v_2 in $\mathbb{R}^2$, respectively. Let $u(v_1, s)$ be a unit vector having the same orientation as half-line s of variable v_1, i.e., $u(v_1, s) = (\cos(\theta_1 + s\frac{360°}{m}), \sin(\theta_1 + s\frac{360°}{m}))$, where θ_1 is the orientation component of v_1. Then constraint $v_1[\![s]\!]v_2$ is satisfiable, if and only if $\vec{v}$ is to the *left* of or *parallel* to $u(v_1, c)$ and $\vec{v}$ is to the *right* of $u(v_1, d)$, i.e., by Lemma 4

$$\det(u(v_1, c), \vec{v}) \geq 0 \quad \text{and} \quad \det(u(v_1, d), \vec{v}) < 0.$$

By expanding the determinant expression according to (2) we obtain a conjunction of two inequalities which are linear in x_1, x_2, y_1, y_2 and nonlinear in θ_1. □

Theorem 6. *CSP($\mathcal{SV}_m$) is in NP.*

Proof. An NP algorithm for CSP($\mathcal{SV}_m$) is obtained by non-deterministically choosing the atomic relations and values for the orientations of the input formula ϕ, and translating each conjunct of ϕ to inequalities by Lemma 5. These inequalities are all linear as the non-linear parts containing θ_i are already instantiated, and can be solved in polynomial time by [Schrijver, 1986]. □

6 A Practical Algorithm for CSP($\mathcal{SV}_m$)

The decision procedure in Theorem 6 is not yet practical as it non-deterministically and non-systematically selects values for orientation variables $\theta_1, \ldots, \theta_n$. In this section we present an improved algorithm that searches for the values of the orientation variables in a systematic manner. The algorithm decides CSP($\mathcal{SV}_m$) and returns a model of the input formula for all $m = 2^r$ where r is a positive integer.

The idea behind the algorithm is based on the following four observations:

1. If we fix orientations $\theta_1, \ldots, \theta_n$ of all variables $v_1, \ldots, v_n$ of an atomic CSP($\mathcal{SV}_m$) instance ϕ, then ϕ can be solved in polynomial time. However, there are m^n possible choices for fixing the orientations.
2. To overcome this exponential complexity in searching for the orientations we use a *duality* between the uncertainty about the orientation and the uncertainty about the direction relation. The next example illustrates this duality.

Example 7. An atomic constraint $v_i[\![1]\!]v_j$ where the orientation of v_i is restricted to $\{135°, 180°, 225°\}$ (see Figure 3a) is satisfiable if and only if constraint $v_i[\![4..7]\!]v_j$ is satisfiable where the orientation of v_i is fixed to $0°$ (see Figure 3b). Here, the uncertainty about the orientation (i.e., $\{135°, 180°, 225°\}$) is transferred to the uncertainty of the direction relation (i.e., $[\![4..7]\!]$). Though solving both constraints is equivalent, the latter can be solved more efficiently, because we do not have to search for the orientations, and convex relations like $[\![4..7]\!]$ allow for efficient reasoning.

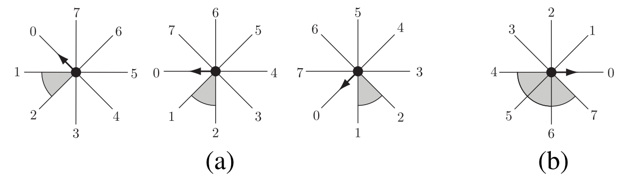


Figure 3: Illustration of duality between uncertainty of orientation and direction relation.

3. By using this duality we can solve a weaker problem efficiently. Parameters that do not even satisfy the weaker problem can therefore be pruned.
4. By using the weaker problem a binary search with pruning can be realized that recursively restricts the domain of orientation variables until a model is found or unsatisfiability is shown.

The four observations are employed in algorithm DECIDESTARVARS (Algorithm 1) which has SEARCH (Algorithm 2) as a subroutine. In what follows, for two variables $v_i = (x_i, y_i, \theta_i)$ and $v_j = (x_j, y_j, \theta_j)$ we will use notations $v_i\, R_{ij}\, v_j$ and $R_{ij}(x_i, y_i, x_j, y_j, \theta_i)$ interchangeably to denote an $\mathcal{SV}_m$ constraint. Note that the orientation variable θ_j is omitted in $R_{ij}(x_i, y_i, x_j, y_j, \theta_i)$, as an $\mathcal{SV}_m$ relation is not dependent on the orientation of the second variable.

Algorithm 1: DECIDESTARVARS(ϕ, m, n)

Input: A CSP($\mathcal{SV}_m$) instance $\phi = \bigwedge_{i \neq j} R_{ij}(x_i, y_i, x_j, y_j, \theta_i)$, granularity $m = 2^r (r \in \mathbb{N})$, and number n of variables.

Output: If ϕ is satisfiable, then a model is returned. Otherwise, fail is returned.

1 **begin**
2 **for each** R_{ij} *in* ϕ **do**
3 Choose an atomic relation $[\![s_{ij}]\!] \in \mathcal{R}$ such that $[\![s_{ij}]\!] \subset R_{ij}$. Then substitute $[\![s_{ij}]\!]$ for R_{ij} in formula ϕ.
4 $\psi \leftarrow \phi$
5 **for each** *conjunct* $[\![s_{ij}]\!](x_i, y_i, x_j, y_j, \theta_i)$ *in* ψ **do**
6 Substitute θ_{ij} for θ_i.
7 **return** SEARCH($\psi, m, n, \Theta_m|_0^m, \ldots, \Theta_m|_0^m$)

We now detail the algorithm: On input ϕ, m, n algorithm DECIDESTARVARS first makes ϕ atomic by picking an atomic relation for each relation in ϕ non-deterministically (lines 2–3). In a deterministic variant this procedure can be realized with a backtracking search.

To enable pruning of the parameter space with the duality property previously mentioned, a new formula ψ is generated by substituting θ_{ij} for θ_i (lines 4–6). Then formula ψ is passed to the subroutine SEARCH, which systematically searches for the values for θ_{ij}. The notation $\mathrm{dom}(\theta_{ij}) = \Theta_m|_a^b$ is used to denote a reduced parameter space of the values for θ_{ij}, where $\mathrm{dom}(\theta_{ij})$ stands for the domain of θ_{ij} and

$$\Theta_m|_a^b := \{k \cdot \tfrac{360°}{m} \mid k = a, a+1, \ldots, b-1\}.$$

Algorithm 2: SEARCH$(\psi, m, n, \Theta_m|_{a_1}^{b_1}, \ldots, \Theta_m|_{a_n}^{b_n})$

Input: An atomic formula $\psi = \bigwedge_{i \neq j} [\![s_{ij}]\!](x_i, y_i, x_j, y_j, \theta_{ij})$, granularity $m = 2^r (r \in \mathbb{N})$, number n of variables, and domain restrictions $\Theta_m|_{a_i}^{b_i}$ for $i = 1, \ldots, n$.

Output: If ψ satisfiable with $\mathrm{dom}(\theta_{ij}) = \Theta_m|_{a_i}^{b_i}$ for $i, j = 1, \ldots, n, i \neq j$, then a model of ψ is returned. Otherwise, fail is returned.

```
1 begin
2     for i ← 1 to n do
3         Θ_m|_{a_i}^{b_i} ← CHOOSE(Θ_m|_{a_i}^{(a_i+b_i)/2}, Θ_m|_{(a_i+b_i)/2}^{b_i})
4     if ψ has a model with dom(θ_ij) = Θ_m|_{a_i}^{b_i} then
5         if b_i − a_i = 1 for i = 1, ..., n then
6             return a model of ψ
7         return SEARCH(ψ, m, n, Θ_m|_{a_1}^{b_1}, ..., Θ_m|_{a_n}^{b_n})
8     return fail
```

For example the orientations values $\{135°, 180°, 225°\}$ in Example 7 is given by $\Theta_8|_4^6$. We note that $\Theta_m|_0^m$ is the complete orientation domain Θ_m.

For brevity of presentation SEARCH is realized in a non-deterministic way using selection (CHOOSE, line 3) and failure (`fail`, line 8). CHOOSE will always return the correct halves of $\Theta_m|_{a_i}^{b_i}, i = 1, \ldots, n$ for which the input formula ψ is satisfiable (line 4). If these do not exist, the algorithm will terminate and return `fail`. The satisfiability check in line 4 prunes the search space in the deterministic variant of algorithm SEARCH. In what follows we prove that this can be done in polynomial time by applying the duality property and solving a conjunction of linear inequalities.

As already observed in Example 7, there is a duality between the uncertainty about orientations and the uncertainty about direction relations, i.e., the disjunction of sectors in Figure 3a is equal to the sector in Figure 3b. In general, the following duality equation holds for two oriented points $A = (x_1, y_1, \theta_1)$ and $B = (x_2, y_2, \theta_2)$ with fixed position (x_1, y_1) of A:

$$\begin{aligned} &\{(x_2, y_2) \in \mathbb{R}^2 \mid \exists \theta_1 \in \Theta_m|_a^b \; [\![s]\!](x_1, y_1, x_2, y_2, \theta_1)\} \\ &= \{(x_2, y_2) \in \mathbb{R}^2 \mid [\![s + a \,..\, s + b]\!](x_1, y_1, x_2, y_2, 0°)\} \end{aligned} \tag{3}$$

Theorem 8. *Finding a model of formula* $\psi := \bigwedge_{i \neq j} [\![s_{ij}]\!](x_i, y_i, x_j, y_j, \theta_{ij})$, *where each* θ_{ij} *is restricted to* $\Theta_m|_{a_i}^{b_i}$ *with* $b_i - a_i \leq m/2, i, j = 1, \ldots, n, i \neq j$, *can be done in polynomial time.*

Proof sketch. We first show that the satisfiability of the stated problem is equivalent to the satisfiability of formula $\psi' := \bigwedge_{i \neq j} [\![s_{ij} + a_i \,..\, s_{ij} + b_i]\!](x_i, y_i, x_j, y_j, 0°)$:

$$\begin{aligned} &\{((x_1, y_1), \ldots, (x_n, y_n)) \in \mathbb{R}^{2n} \mid \\ &\qquad \exists \theta_{ij} \in \Theta_m|_{a_i}^{b_i} \bigwedge_{i \neq j} [\![s_{ij}]\!](x_i, y_i, x_j, y_j, \theta_{ij})\} \\ &= \{((x_1, y_1), \ldots, (x_n, y_n)) \in \mathbb{R}^{2n} \mid \\ &\qquad \bigwedge_{i \neq j} \exists \theta_{ij} \in \Theta_m|_{a_i}^{b_i} \; [\![s_{ij}]\!](x_i, y_i, x_j, y_j, \theta_{ij})\} \\ &= \{((x_1, y_1), \ldots, (x_n, y_n)) \in \mathbb{R}^{2n} \mid \\ &\qquad \bigwedge_{i \neq j} [\![s_{ij} + a_i \,..\, s_{ij} + b_i]\!](x_i, y_i, x_j, y_j, 0°)\} \end{aligned}$$

The first equality is due the fact that each θ_{ij} appears in exactly one conjunct and can therefore be pulled into the conjunction. The second equality follows from equation (3). Now we show that ψ' is solvable in polynomial time. Solving ψ' is equivalent to solving a conjunction of inequalities, because all relations in ψ' are convex ($b_i - a_i \leq m/2$) such that each conjunct of ψ' can be translated to inequalities by Lemma 5. These inequalities are all linear as the non-linear parts containing θ_i evaluate to constants, and can be solved in polynomial time by [Schrijver, 1986]. □

Corollary 9. *The satisfiability check in line 4 of Algorithm 2 can be done in polynomial time.*

Now we prove that DECIDESTARVARS is sound and complete. The next lemma states that satisfiability of ψ is a weaker problem, i.e., a necessary condition for satisfiability of the input formula ϕ.

Lemma 10. *Let* $\phi := \bigwedge_{i \neq j} [\![s_{ij}]\!](x_i, y_i, x_j, y_j, \theta_i)$ *and* $\psi := \bigwedge_{i \neq j} [\![s_{ij}]\!](x_i, y_i, x_j, y_j, \theta_{ij})$ *be two formulas with* $\mathrm{dom}(\theta_i) = \mathrm{dom}(\theta_{ij}) = \Theta_m|_{a_i}^{b_i}$ *for* $i, j = 1, \ldots, n, i \neq j$. *Then* ϕ *is satisfiable only if* ψ *is satisfiable.*

Proof. Let a model M of ϕ be given. Then we vacuously obtain a model of ψ which is same as M except for the fact that the valuation of θ_i is extended to θ_{ij}. □

After some recursive application of SEARCH each restricted orientation domain $\Theta_m|_{a_i}^{b_i}$ consists of only one element. Then a model of ψ induces a model of ϕ:

Lemma 11. *Let* ϕ *and* ψ *be defined as in Lemma 10. If all* $\Theta_m|_{a_i}^{b_i}$ *consists of only one element, i.e.,* $b_i - a_i = 1$ *for* $i = 1, \ldots, n$, *then* ϕ *is satisfiable if* ψ *is satisfiable and a model of* ψ *can be identified with a model of* ϕ.

Proof. Because $\Theta_m|_{a_i}^{b_i}$ consists of only one element, say t_i, all θ_{ij} for all $j = 1, \ldots, n$ are evaluated to the same value t_i. Thus, given a model of ψ, the assignment of the value t_i to θ_i while preserving the values for x_i, y_i for each i, yields a model of ϕ. □

Theorem 12. DECIDESTARVARS *is a sound and complete algorithm.*

Proof. For the *soundness* assume that on input ϕ, m and n DECIDESTARVARS returns a model. Then, according to line 5–6 in Algorithm 2, ψ is satisfiable with $\mathrm{dom}(\theta_{ij}) = \Theta_m|_{a_i}^{b_i}$ with $b_i - a_i = 1$ for $i = 1, \ldots, n$. Then by applying Lemma 11 we obtain a model of ϕ.

For the *completeness* assume that DECIDESTARVARS returns fail. Then ψ is not satisfiable, because SEARCH implements a complete search for finding the values for θ_{ij} such that there is a model of ψ. It follows that ϕ is not satisfiable by Lemma 10. □

7 Evaluation

We evaluated DECIDESTARVARS for 100 random atomic CSP($\mathcal{SV}_m$) instances with varying m and n, and have recorded the average computing time in seconds (the variance is shown in parentheses). For solving systems of linear inequalities we used function `linprog` from the MATLAB Optimization Toolbox. The evaluation was done on a machine with Intel® Core™2 E6700 CPU and 4 GB RAM.

n	m = 4	8	16	32
3	0.27 (0.02)	0.62 (0.15)	1.25 (0.91)	1.73 (2.24)
4	0.64 (0.15)	1.15 (0.50)	2.01 (1.28)	2.63 (2.60)
5	1.06 (0.34)	1.66 (1.58)	2.56 (4.57)	4.35 (14.99)
6	2.55 (0.00)	3.16 (2.68)	4.27 (12.14)	6.10 (34.58)
7	6.83 (0.01)	7.55 (0.01)	8.30 (3.32)	8.76 (7.19)

From the table we conclude that the increase of computing time in the size of granularity m is only logarithmic in average case. This efficiency, which is obtained by the pruning step integrated in the algorithm, can be utilized to approximate relative directions with a high resolution.

However, as the algorithmic complexity suggests, the computing time increases exponentially in the number n of variables. Thus, the proposed algorithm is suitable for applications which allow for precomputation or which involve only a limited number of objects.

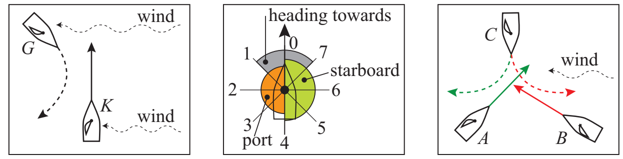
8 Application Example

Mossakowski and Moratz [2012] present representation of navigation regulations as a relevant application domain for directional calculi. We pick up the example from this paper which is set in the context of sea navigation. Maritime traffic regulations issued by the International Maritime Organization (IMO) comprise the following rule:

> When two sailing vessels are approaching one another, as to involve the risk of collision, one of them shall keep out of the way of the other as follows: (i) when each has the wind on a different side, the vessel which has the wind on the port side shall keep out of the way of the other. (Rule 12 i, IMO)

See Figure 4a for an illustration of the rule in which vessel G has to give way and vessel K has to keep course, using the common avoidance behavior of turning away from the vessel that has right of way. Vessel G aims to pass behind the stern of vessel K, thus turning towards it. Considering rule and avoidance pattern are stated for exactly two vessels, an important question is: does this rule also handle a 3-ship encounter? Or can it lead to contradicting recommendations? To answer this question we model the rule in $\mathcal{SV}_8$, mapping natural language terms to qualitative spatial relations as shown in Figure 4b. Assuming the same wind direction for both vessels, we obtain the following conjunction of qualitative relations:

$$\phi(K,G,W) := \begin{cases} K\,[\![0..4]\!]W & \text{(wind on port)} \\ \wedge\, G\,[\![4..0]\!]W & \text{(wind on starboard)} \\ \wedge\, G\,[\![7..1]\!]K & (G \text{ heading towards } K) \\ \wedge\, K\,[\![7..1]\!]G & (K \text{ heading towards } G) \end{cases}$$



(a) (b) (c)

Figure 4: (a) G must give way to K according to Rule 12 (ii). (b) qualitative modeling with $\mathcal{SV}_m$. (c) Rule conflict for multi-vessel encounter, vessel C has to turn port and starboard at the same time.

Additionally, vessel G should give way by turning to starboard, if vessel K approaches from starboard, and turning to port, otherwise. We can thus describe the situations that lead to mutually exclusive turning actions by formulas α and β:

$$\begin{aligned} \alpha(K,G,W) &= \phi(K,G,W) \wedge G[\![0..4]\!]K, \\ \beta(K,G,W) &= \phi(K,G,W) \wedge G[\![4..0]\!]K \end{aligned}$$

To answer our question about a 3-ship encounter we construct a CSP with variables for vessels A, B, C and wind W:

$$\alpha(A,C,W) \wedge \beta(B,C,W) \tag{4}$$

If formula (4) is satisfiable, then there exists a configuration that requires two mutually exclusive commands to be carried out and the rule is known to not generalize to 3-ship encounters. Our sound decision method determines satisfiability of formula (4) and it outputs a configuration from which we generated Figure 4c, showing a conflict for vessel C.

One could approach the task with constraint propagation using relation composition as suggested by [Mossakowski and Moratz, 2012]. This outputs, however, false-positives and cannot compute a realization. Thus one has to check each output manually which is an infeasible task. Our reasoning method is thus the superior approach to the task. For more information on the application of StarVars in the sea navigation domain we refer to [Kreutzmann *et al.*, 2013].

9 Summary and Conclusions

We have developed a spatial representation, StarVars, which augments cardinal direction relations to represent relative directional knowledge. By introducing orientation variables we are able to apply computationally cheaper decision procedures for cardinal directions to the hard problem of handling directional knowledge. We gave an NP decision procedure for StarVars, and therefore it can replace existing directional relation languages for which no effective decision procedures are available today. Additionally, our algorithm can determine a model of consistent CSP instances, which is valuable for many applications that require a visual presentation.

Future works aims to improve our decision procedure, in particular to counter-act the exponential growth with respect to the number of variables using heuristics.

Acknowledgements

The authors thank the anonymous reviewers for helpful feedback. Financial support by DFG (SFB/TR 8 Spatial Cognition, project R3-[Q-Shape]) is acknowledged.

References

[Aiello *et al.*, 2007] Marco Aiello, Ian Pratt-Hartmann, and Johan Benthem. What is spatial logic? In Marco Aiello, Ian Pratt-Hartmann, and Johan Benthem, editors, *Handbook of Spatial Logics*, pages 1–11. Springer Netherlands, 2007.

[Cohn and Renz, 2008] Anthony G. Cohn and Jochen Renz. Chapter 13 qualitative spatial representation and reasoning. In Vladimir Lifschitz Frank van Harmelen and Bruce Porter, editors, *Handbook of Knowledge Representation*, volume 3 of *Foundations of Artificial Intelligence*, pages 551 – 596. Elsevier, 2008.

[Freksa, 1992] C. Freksa. Using orientation information for qualitative spatial reasoning. In *Theories and Methods of Spatio-Temporal Reasoning in Geographic Space. International Conference GIS — From Space to Territory*, pages 162–178. Springer, 1992.

[Kreutzmann *et al.*, 2013] Arne Kreutzmann, Diedrich Wolter, Frank Dylla, and Jae Hee Lee. Towards safe navigation by formalizing navigation rules. *TransNav, the International Journal on Marine Navigation and Safety of Sea Transportation*, 7(2), June 2013.

[Ligozat and Renz, 2004a] G. Ligozat and J. Renz. What is a qualitative calculus? A general framework. In C. Zhang, H. W. Guesgen, and W. K. Yeap, editors, *PRICAI 2004: Trends in Artificial Intelligence—8th Pacific Rim International Conference on Artificial Intelligence*, pages 53–64. Springer, 2004.

[Ligozat and Renz, 2004b] Gérard Ligozat and Jochen Renz. What is a qualitative calculus? A general framework. In Chengqi Zhang, Hans Werner Guesgen, and Wai K. Yeap, editors, *PRICAI 2004: Trends in Artificial Intelligence—8th Pacific Rim International Conference on Artificial Intelligence*, pages 53–64. Springer, 2004.

[Ligozat, 1993] G. Ligozat. Qualitative triangulation for spatial reasoning. In A. U. Frank and I. Campari, editors, *Spatial Information Theory: A Theoretical Basis for GIS (COSIT'93)*, volume 716 of *Lecture Notes in Computer Science*, pages 54–68. Springer, 1993.

[Lücke *et al.*, 2008] D. Lücke, T. Mossakowski, and D. Wolter. Qualitative reasoning about convex relations. In C. Freksa, S. N. Newcombe, P. Gärdenfors, and Stefan Wölfl, editors, *Spatial Cognition VI*, volume 5248 of *Lecture Notes in Computer Science*, pages 426–440. Springer, 2008.

[Moratz and Ragni, 2007] R. Moratz and M. Ragni. Qualitative spatial reasoning about relative point position. *Journal of Visual Languages and Computing*, 19(1):75–98, 2007.

[Moratz, 2006] R. Moratz. Representing relative direction as a binary relation of oriented points. In *Proceedings of ECAI*, pages 407–411, 2006.

[Mossakowski and Moratz, 2012] Till Mossakowski and Reinhard Moratz. Qualitative reasoning about relative direction of oriented points. *Artificial Intelligence*, 180–181:34–45, 2012.

[Opatrny, 1979] J. Opatrny. Total ordering problems. *SIAM Journal on Computing*, 8(1):111–114, 1979.

[Renz and Mitra, 2004] J. Renz and D. Mitra. Qualitative direction calculi with arbitrary granularity. In *PRICAI 2004: Trends in Artificial Intelligence*. Springer, 2004.

[Renz and Nebel, 2007] J. Renz and B. Nebel. Qualitative spatial reasoning using constraint calculi. In M. Aiello, Ian E. Pratt-Hartmann, and Johan F.A.K. van Benthem, editors, *Handbook of Spatial Logics*, pages 161–215. Springer, 2007.

[Schrijver, 1986] Alexander Schrijver. *Theory of linear and integer programming*. John Wiley & Sons, Inc., New York, NY, USA, 1986.

[Wolter and Lee, 2010] D. Wolter and J. H. Lee. Qualitative reasoning with directional relations. *Artificial Intelligence*, 174(18):1498 – 1507, 2010.