Combining cardinal direction relations and relative orientation relations in Qualitative Spatial Reasoning*

Amar Isli, Volker Haarslev and Ralf Möller

Fachbereich Informatik, Universität Hamburg,
Vogt-Kölln-Strasse 30, D-22527 Hamburg, Germany
{isli,haarslev,moeller}@informatik.uni-hamburg.de

Abstract. Combining different knowledge representation languages is one of the main topics in Qualitative Spatial Reasoning (QSR). This allows the combined languages to compensate each other's representational deficiencies, and is seen as an asnwer to the emerging demand from real applications, such as Geographical Information Systems (GIS), robot navigation, or shape description, for the representation of more specific knowledge than is allowed by each of the languages taken separately. Knowledge expressed in such a combined language decomposes then into parts, or components, each expressed in one of the combined languages. Reasoning internally within each component of such knowledge involves only the language the component is expressed in, which is not new. The challenging question is to come with methods for the interaction of the different components of such knowledge. With these considerations in mind, we propose a calculus, $c\mathcal{COA}$, combining, thus more expressive than each of, two calculi well-known in QSR: Frank's cardinal direction calculus, $\mathcal{CDA}$, and a coarser version, $\mathcal{ROA}$, of Freksa's relative orientation calculus. An original constraint propagation procedure, *PcS4c+()*, for $c\mathcal{COA}$-CSPs is presented, which aims at (1) achieving path consistency (*Pc*) for the $\mathcal{CDA}$ projection; (2) achieving strong 4-consistency (*S4c*) for the $\mathcal{ROA}$ projection; and (3) more (+) (the "+" consists of the implementation of the interaction between the two combined calculi). Dealing with the first two points is not new, and involves mainly the $\mathcal{CDA}$ composition table and the $\mathcal{ROA}$ composition table, which can be found in, or derived from, the literature. The originality of the propagation algorithm comes from the last point. Two tables, one for each of the two directions $\mathcal{CDA}$-to-$\mathcal{ROA}$ and $\mathcal{ROA}$-to-$\mathcal{CDA}$, capturing the interaction between the two kinds of knowledge, are defined, and used by the algorithm. The importance of taking into account the interaction is shown with a real example providing an inconsistent knowledge base, whose inconsistency (a) cannot be detected by reasoning separately about each of the two components of the knowledge, just because, taken separately, each is consistent, but (b) is detected by the proposed algorithm, thanks to the interaction knowledge propagated from each of the two compnents to the other.

Key words: Qualitative spatial reasoning, Cardinal directions, Relative orientation, Constraint satisfaction, Path consistency, Strong 4-consistency.

* This work was partly supported by the DFG project "*Description Logics and Spatial reasoning*", under grant *NE 279/8-1*.

1 Introduction

Reasoning about orientation has been, for about a decade now, one of the main aspects focussed on in Qualitative Spatial Reasoning (QSR). A possible explanation stems from the large number of real applications in need for a formalism for representing and reasoning about orientation; among these, we have robot navigation, Geographical Information Systems (GIS), and shape description. The reader is referred to [3] for a survey article on the different representation techniques, and the different aspects dealt with, in QSR.

Two important, and widely known, calculi for the representation and processing of orientation are the calculus of cardinal directions, $\mathcal{CDA}$, developed by Frank [7, 8], and the relative orientation calculus developed by Freksa [9, 10]. The former uses a global, south-north/west-east reference frame, and represents knowledge as binary relations on (pairs of) 2D points. The latter allows for the representation of relative knowledge as ternary relations on (triples of) 2D points. Both kinds of knowledge are of particular importance, especially in GIS.

The research in QSR has reached a point where the need for combining different aspects, such as, in the present work, global orientation [7, 8] and relative orientation [9, 10], and different techniques, such as, also in the present work, path consistency and strong 4-consistency, is necessary in order to face the increasing and often challenging demand coming from real applications.

The aim of this work is to look at the importance of combining the two orientation calculi mentioned above. Considered separately, Frank's calculus [7, 8], $\mathcal{CDA}$, represents knowledge such as "Hamburg is north-west of Berlin", whereas Freksa's relative orientation calculus [9, 10] represents knowledge such as "You see the main train station on your left when you walk down to the cinema from the university". We propose a calculus, $c\mathcal{COA}$, combining $\mathcal{CDA}$ and a coarser version, $\mathcal{ROA}$, of Freksa's calculus. $c\mathcal{COA}$ allows for more expressiveness than each of the combined calculi, and represents, within the same base, knowledge such as the one in the following example.

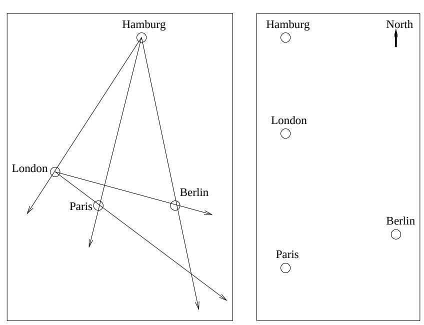

Fig. 1. A model for the $\mathcal{ROA}$ component (left), and a model for the $\mathcal{CDA}$ component (right), of the knowledge in Example 1.

Example 1 *Consider the following knowledge on four cities, Berlin, Hamburg, London and Paris:*

1. *viewed from Hamburg, Berlin is to the left of Paris, Paris is to the left of London, and Berlin is to the left of London;*
2. *viewed from London, Berlin is to the left of Paris;*
3. *Hamburg is to the north of Paris, and north-west of Berlin; and*
4. *Paris is to the south of London.*

The first two sentences express the $\mathcal{ROA}$ component of the knowledge (relative orientation relations on triples of the four cities), whereas the other two express the $\mathcal{CDA}$ component of the knowledge (cardinal direction relations on pairs of the four cities).[1] *Considered separately, each of the two components is consistent, in the sense that one can find an assignment of physical locations to the cities that satisfies all the constraints of the component —see the illustration in Figure 1. However, considered globally, the knowledge is clearly inconsistent.*

Example 1 clearly shows that reasoning about combined knowledge consisting of an $\mathcal{ROA}$ component and a $\mathcal{CDA}$ component, e.g., checking its consistency, does not reduce to a matter of reasoning about each component separately —reasoning separately about each component in the case of Example 1 shows two components that are both consistent, whereas the conjunction of the knowledge in the two components is inconsistent. As a consequence, the interaction between the two kinds of knowledge has to be handled. With this in mind, we propose a constraint propagation procedure, *PcS4c+()*, for $c\mathcal{COA}$-CSPs, which aims at:

1. achieving path consistency (*Pc*) for the $\mathcal{CDA}$ projection;
2. achieving strong 4-consistency (*S4c*) for the $\mathcal{ROA}$ projection; and
3. more (+).

The procedure does more than just achieving path consistency for the $\mathcal{CDA}$ projection, and strong 4-consistency for the $\mathcal{ROA}$ projection. It implements as well the interaction between the two combined calculi. For this purpose:

1. The procedure makes use, on the one hand, of an augmented composition table of the $\mathcal{CDA}$ calculus:
   (a) the table records, for each pair (r, s) of $\mathcal{CDA}$ atoms, the standard composition, $r \circ s$, of r and s, which is not new, and can be found in the literature [7, 8, 19]; and
   (b) more importantly, the table records the $\mathcal{CDA}$-to-$\mathcal{ROA}$ interaction, by providing, for each pair (r, s) of $\mathcal{CDA}$ atoms, the most specific $\mathcal{ROA}$ relation, $r \otimes s$, such that, for all x, y, z, the conjunction $r(x, y) \wedge s(y, z)$ logically implies $(r \otimes s)(x, y, z)$.
2. On the other hand, the procedure makes of a table for the $\mathcal{ROA}$-to-$\mathcal{CDA}$ interaction, providing, for each $\mathcal{ROA}$ atom t, the constraints it imposes on the $\mathcal{CDA}$ relations on the different pairs of the three arguments.

The procedure is, to the best of our knowledge, original.

The rest of the paper is organised as follows. Section 2 provides some background on constraint satisfaction problems (CSPs), on constraint matrices and on relation

[1] Two cardinal direction calculi, to be explained later, are known from Frank's work [7, 8]; we assume in this example the one in Figure 2(right).

algebras. Section 3 presents a quick overview of Frank's cardinal directions calculus [7, 8], and of Freksa's relative orientation calculus [9, 10]. Section 4 defines a relative orientation calculus, $\mathcal{ROA}$, which is a caorser version of Freksa's [9, 10]. Reasoning in the combined language of $\mathcal{CDA}$ relations and $\mathcal{ROA}$ relations is dealt with in detail in Section 5; in particular, the section presents the $\mathcal{CDA}$-to-$\mathcal{ROA}$ and the $\mathcal{ROA}$-to-$\mathcal{CDA}$ interaction tables, as well as the constraint propagation algorithm *PcS4c+()*, both alluded to above. Section 6 summarises the work.

2 Constraint satisfaction problems

A constraint satisfaction problem (CSP) of order n consists of:

1. a finite set of n variables, $x_1, \ldots, x_n$;
2. a set U (called the universe of the problem); and
3. a set of constraints on values from U which may be assigned to the variables.

An m-ary constraint is of the form $R(x_{i_1}, \cdots, x_{i_m})$, and asserts that the m-tuple of values assigned to the variables $x_{i_1}, \cdots, x_{i_m}$ must lie in the m-ary relation R (an m-ary relation over the universe U is any subset of U^m). An m-ary CSP is one of which the constraints are m-ary constraints. We will be concerned exclusively with binary CSPs and ternary CSPs.

For any two binary relations R and S, $R \cap S$ is the intersection of R and S, $R \cup S$ is the union of R and S, $R \circ S$ is the composition of R and S, and $R^\smile$ is the converse of R; these are defined as follows:

$$
\begin{aligned}
R \cap S &= \{(a,b) : (a,b) \in R \text{ and } (a,b) \in S\}, \\
R \cup S &= \{(a,b) : (a,b) \in R \text{ or } (a,b) \in S\}, \\
R \circ S &= \{(a,b) : \text{for some } c, (a,c) \in R \text{ and } (c,b) \in S\}, \\
R^\smile &= \{(a,b) : (b,a) \in R\}.
\end{aligned}
$$

Three special binary relations over a universe U are the empty relation $\emptyset$ which contains no pairs at all, the identity relation $\mathcal{I}_U^b = \{(a,a) : a \in U\}$, and the universal relation $\top_U^b = U \times U$.

Composition and converse for binary relations were introduced by De Morgan [4, 5]. Isli and Cohn [15, 16] extended the two operations to ternary relations; furthermore, they introduced for ternary relations the operation of rotation, which is not needed for binary relations. For any two ternary relations R and S, $R \cap S$ is the intersection of R and S, $R \cup S$ is the union of R and S, $R \circ S$ is the composition of R and S, $R^\smile$ is the converse of R, and $R^\frown$ is the rotation of R; these are defined as follows:

$$
\begin{aligned}
R \cap S &= \{(a,b,c) : (a,b,c) \in R \text{ and } (a,b,c) \in S\}, \\
R \cup S &= \{(a,b,c) : (a,b,c) \in R \text{ or } (a,b,c) \in S\}, \\
R \circ S &= \{(a,b,c) : \text{for some } d, (a,b,d) \in R \text{ and } (a,d,c) \in S\}, \\
R^\smile &= \{(a,b,c) : (a,c,b) \in R\}, \\
R^\frown &= \{(a,b,c) : (c,a,b) \in R\}.
\end{aligned}
$$

Three special ternary relations over a universe U are the empty relation $\emptyset$ which contains no triples at all, the identity relation $\mathcal{I}_U^t = \{(a,a,a) : a \in U\}$, and the universal relation $\top_U^t = U \times U \times U$.

2.1 Constraint matrices

A binary constraint matrix of order n over U is an $n \times n$-matrix, say $\mathcal{B}$, of binary relations over U verifying the following:

$$
\begin{array}{ll}
(\forall i \leq n)(\mathcal{B}_{ii} \subseteq \mathcal{I}_U^b) & \text{(the diagonal property)}, \\
(\forall i,j \leq n)(\mathcal{B}_{ij} = (\mathcal{B}_{ji})^{\smile}) & \text{(the converse property)}.
\end{array}
$$

A binary CSP P of order n over a universe U can be associated with the following binary constraint matrix, denoted $\mathcal{B}^P$:

1. Initialise all entries to the universal relation: $(\forall i,j \leq n)((\mathcal{B}^P)_{ij} \leftarrow \top_U^b)$
2. Initialise the diagonal elements to the identity relation: $(\forall i \leq n)((\mathcal{B}^P)_{ii} \leftarrow \mathcal{I}_U^b)$
3. For all pairs (x_i, x_j) of variables on which a constraint $(x_i, x_j) \in R$ is specified: $(\mathcal{B}^P)_{ij} \leftarrow (\mathcal{B}^P)_{ij} \cap R, (\mathcal{B}^P)_{ji} \leftarrow ((\mathcal{B}^P)_{ij})^{\smile}$.

A ternary constraint matrix of order n over U is an $n \times n \times n$-matrix, say $\mathcal{T}$, of ternary relations over U verifying the following:

$$
\begin{array}{ll}
(\forall i \leq n)(\mathcal{T}_{iii} \subseteq \mathcal{I}_U^t) & \text{(the identity property)}, \\
(\forall i,j,k \leq n)(\mathcal{T}_{ijk} = (\mathcal{T}_{ikj})^{\smile}) & \text{(the converse property)}, \\
(\forall i,j,k \leq n)(\mathcal{T}_{ijk} = (\mathcal{T}_{kij})^{\frown}) & \text{(the rotation property)}.
\end{array}
$$

A ternary CSP P of order n over a universe U can be associated with the following ternary constraint matrix, denoted $\mathcal{T}^P$:

1. Initialise all entries to the universal relation: $(\forall i,j,k \leq n)((\mathcal{T}^P)_{ijk} \leftarrow \top_U^t)$
2. Initialise the diagonal elements to the identity relation: $(\forall i \leq n)((\mathcal{T}^P)_{iii} \leftarrow \mathcal{I}_U^t)$
3. For all triples (x_i, x_j, x_k) of variables on which a constraint $(x_i, x_j, x_k) \in R$ is specified:
   $(\mathcal{T}^P)_{ijk} \leftarrow (\mathcal{T}^P)_{ijk} \cap R$, $(\mathcal{T}^P)_{ikj} \leftarrow ((\mathcal{T}^P)_{ijk})^{\smile}$,
   $(\mathcal{T}^P)_{jki} \leftarrow ((\mathcal{T}^P)_{ijk})^{\frown}$, $(\mathcal{T}^P)_{jik} \leftarrow ((\mathcal{T}^P)_{jki})^{\smile}$,
   $(\mathcal{T}^P)_{kij} \leftarrow ((\mathcal{T}^P)_{jki})^{\frown}$, $(\mathcal{T}^P)_{kji} \leftarrow ((\mathcal{T}^P)_{kij})^{\smile}$.

We make the assumption that, unless explicitly specified otherwise, a CSP is given as a constraint matrix.

2.2 Strong k-consistency, refinement

Let P be a CSP of order n, V its set of variables and U its universe. An instantiation of P is any n-tuple $(a_1, a_2, \ldots, a_n)$ of U^n, representing an assignment of a value to each variable. A consistent instantiation is an instantiation $(a_1, a_2, \ldots, a_n)$ which is a solution:

- If P is a binary CSP: $(\forall i,j \leq n)((a_i, a_j) \in (\mathcal{B}^P)_{ij})$
- If P is a ternary CSP: $(\forall i,j,k \leq n)((a_i, a_j, a_k) \in (\mathcal{T}^P)_{ijk})$

P is consistent if it has at least one solution; it is inconsistent otherwise. The consistency problem of P is the problem of verifying whether P is consistent.

Let $V' = \{x_{i_1}, \ldots, x_{i_j}\}$ be a subset of V. The sub-CSP of P generated by V', denoted $P_{|V'}$, is the CSP with V' as the set of variables, and whose constraint matrix is obtained by projecting the constraint matrix of P onto V':

- If P is a binary CSP then: $(\forall k, l \leq j)((\mathcal{B}^{P_{|V'}})_{kl} = (\mathcal{B}^P)_{i_k i_l})$
- If P is a ternary CSP then: $(\forall k, l, m \leq j)((\mathcal{T}^{P_{|V'}})_{klm} = (\mathcal{T}^P)_{i_k i_l i_m})$

P is k-consistent [11, 12] if for any subset V' of V containing $k-1$ variables, and for any variable $X \in V$, every solution to $P_{|V'}$ can be extended to a solution to $P_{|V' \cup \{X\}}$. P is strongly k-consistent if it is j-consistent, for all $j \leq k$.

1-consistency, 2-consistency and 3-consistency correspond to node-consistency, arc-consistency and path-consistency, respectively [20, 21]. Strong n-consistency of P corresponds to what is called global consistency in [6]. Global consistency facilitates the important task of searching for a solution, which can be done, when the property is met, without backtracking [12].

A refinement of P is a CSP P' with the same set of variables, and such that

- $(\forall i, j)((\mathcal{B}^{P'})_{ij} \subseteq (\mathcal{B}^P)_{ij})$, in the case of binary CSPs.
- $(\forall i, j, k)((\mathcal{T}^{P'})_{ijk} \subseteq (\mathcal{T}^P)_{ijk})$, in the case of ternary CSPs.

2.3 Relation algebras

The reader is referred to [22, 17] for the definition of a binary Relation Algebra (RA), and to [16] for the definition of a ternary RA. Of particular interest to this work are:

1. binary RAs of the form $\langle \mathcal{A}, \oplus, \odot, ^-, \bot, \top, \circ, ^\smile, \mathcal{I} \rangle$, where $\mathcal{A}$ is a non empty finite set, and $\circ$ and $^\smile$ are the operations of composition and converse, respectively; and
2. ternary RAs of the form $\langle \mathcal{A}, \oplus, \odot, ^-, \bot, \top, \circ, ^\smile, ^\frown, \mathcal{I} \rangle$, where $\mathcal{A}$ is a non empty finite set, and $\circ$, $^\smile$ and $^\frown$ are the operations of composition, converse and rotation, respectively.

3 Existing orientation calculi

Some background on existing orientation calculi is in order.

3.1 Frank's cardinal directions calculi

Frank's models of cardinal directions in 2D [7, 8] are illustrated in Figure 2. They use a partition of the plane into regions determined by lines passing through a reference object, say S. Depending on the region a point P belongs to, we have $No(P, S)$, $NE(P, S)$, $Ea(P, S)$, $SE(P, S)$, $So(P, S)$, $SW(P, S)$, $We(P, S)$, $NW(P, S)$, or $Eq(P, S)$, corresponding, respectively, to the position of P relative to S being *north*, *north-east*,

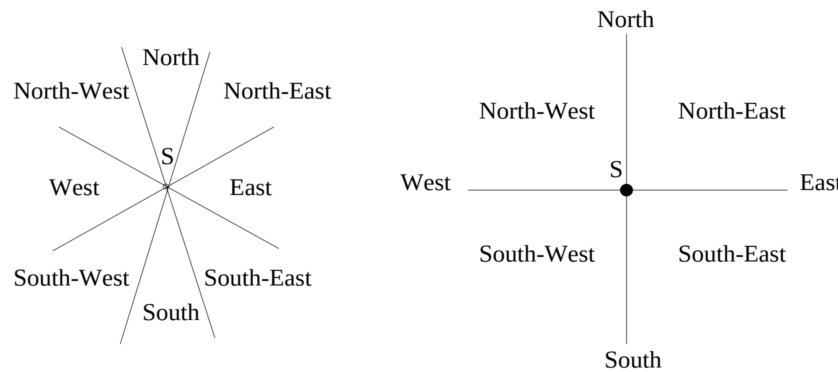

Fig. 2. Frank's cone-shaped (left) and projection-based (right) models of cardinal directions.

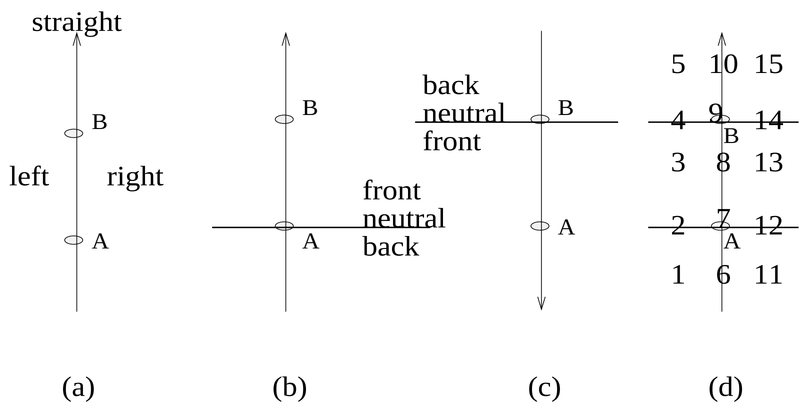

Fig. 3. The partition of the universe of 2D positions on which is based the relative orientation calculus in [9, 23].

east, *south-east*, *south*, *south-west*, *west*, *north-west*, or *equal*. Each of the two models can thus be seen as a binary RA, with nine atoms. Both use a global, *west-east/south-north*, reference frame. We focus our attention on the projection-based model (Figure 2(right)), which has been assessed as being cognitively more adequate [7, 8] (cognitive adequacy of spatial orientation models is discussed in [9, 10]).

3.2 Freksa's relative orientation calculus

A well-known model of relative orientation of 2D points is the calculus defined by Freksa [9], and developed further by Zimmermann and Freksa [23]. The calculus corresponds to a specific partition, into 15 regions, of the plane, determined by a parent object, say A, and a reference object, say B (Figure 3(d)). The partition is based on the following:

1. the *left/straight/right* partition of the plane determined by an observer placed at the parent object and looking in the direction of the reference object (Figure 3(a));

2. the *front/neutral/back* partition of the plane determined by the same observer (Figure 3(b)); and
3. the similar *front/neutral/back* partition of the plane obtained when we swap the roles of the parent object and the reference object (Figure 3(c)).

Combining the three partitions (a), (b) and (c) of Figure 3 leads to the partition of the universe of 2D positions on which is based the calculus in [9, 23] (Figure 3(d)).

4 A new relative orientation calculus

Frank's model of cardinal directions uses a global, west-east/south-north, reference frame; its use and importance in GIS are well-known. Freksa's calculus is more suited for the description of a configuration of 2D points (a spatial scene) relative to one another. Combining the cardinal directions with Freksa's calculus would lead to more expressiveness than allowed by each of the combined calculi, so that one would then be able to represent, within the same base, knowledge such as the one in the 4-sentence example provided in the introduction.

The coarser relative orientation calculus can be obtained from Freksa's calculus by ignoring, in the construction of the partition of the plane determined by a parent object and a reference object (Figure 3(d)), the two *front/neutral/back* partitions (Figure 3(b-c)). In other words, we consider only the *left/straight/right* partition (Figure 3(a)). The final situation is depicted in Figure 4, where A and B are the parent object and the reference object, respectively:

1. Figure 4(b-c) depicts the general case, corresponding to the parent object and the reference object being distinct from each other: the system is based on the *left/straight/right* partition of the plane determined by the directed straight line joining the parent object to the reference object (Figure 4(b)); this general-case partition leads to 7 regions (Figure 4(c)), numbered from 2 to 8: these 7 regions correspond to 7 of the nine atoms of the calculus, which we refer to as lr (to the $\underline{l}$eft of the $\underline{r}$eference object), bp ($\underline{b}$ehind the $\underline{p}$arent object), cp ($\underline{c}$oincides with the $\underline{p}$arent object), bw ($\underline{b}$et$\underline{w}$een the parent object and the reference object), cr ($\underline{c}$oincides with the $\underline{r}$eference object), br ($\underline{b}$ehind the $\underline{r}$eference object), and rr (to the $\underline{r}$ight of the $\underline{r}$eference object).
2. Figure 4(a) illustrates the degenerate case, corresponding to equality of the parent object and the reference object. The two regions, corresponding, respectively, to the primary object coinciding with the parent object and the reference object, and to the primary object distinct from the parent object and the reference object, are numbered 0 and 1. The corresponding atoms of the calculus will be referred to as de ($\underline{d}$egenerate $\underline{e}$qual) and dd ($\underline{d}$egenerate $\underline{d}$istinct).

From now on, we refer to the cardinal directions calculus as $\mathcal{CDA}$ (Cardinal Directions Algebra), and to the coarser version of Freksa's relative orientation calculus as $\mathcal{ROA}$ (Relative Orientation Algebra). A $\mathcal{CDA}$ (resp. $\mathcal{ROA}$) relation is any subset of the set of all $\mathcal{CDA}$ (resp. $\mathcal{ROA}$) atoms. A $\mathcal{CDA}$ (resp. $\mathcal{ROA}$) relation is said to be atomic if it contains one single atom (a singleton set); it is said to be the $\mathcal{CDA}$ (resp. $\mathcal{ROA}$) universal relation if it contains all the $\mathcal{CDA}$ (resp. $\mathcal{ROA}$) atoms. When no confusion raises, we may omit the brackets in the representation of an atomic relation.

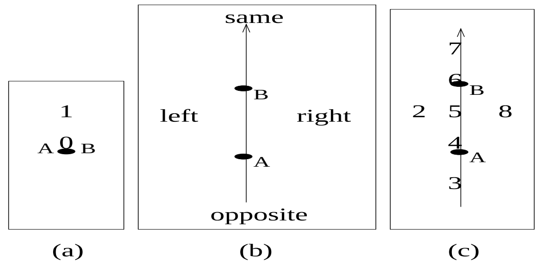

Fig. 4. The partition of the universe of 2D positions on which is based the $\mathcal{ROA}$ calculus.

5 Reasoning about combined knowledge of $\mathcal{CDA}$ relations and $\mathcal{ROA}$ relations

We start now the main part of the paper, i.e., the representation of knowledge about 2D points as a combined conjunction of:

1. $\mathcal{CDA}$ relations on (pairs of) the objects, on the one hand; and
2. $\mathcal{ROA}$ relations on (triples of) the objects, on the other hand.

More importantly, we deal with the issue of reasoning about such a combined knowledge. We first present for each of the combined calculi, $\mathcal{CDA}$ and $\mathcal{ROA}$:

1. tables recording the internal reasoning: the tables of converse and composition for $\mathcal{CDA}$, which can be found in the literature [7, 8, 19]; and the tables of converse, rotation and composition for $\mathcal{ROA}$, which can be derived from the work in [15, 16]; and
2. a table for the interaction with the other calculus: a $\mathcal{CDA}$-to-$\mathcal{ROA}$ interaction table, recording the $\mathcal{ROA}$ knowledge inferred from $\mathcal{CDA}$ knowledge; and an $\mathcal{ROA}$-to-$\mathcal{CDA}$ interaction table, recording the $\mathcal{CDA}$ knowledge inferred from $\mathcal{ROA}$ knowledge.

We then give a quick presentation of what is already known in the literature: CSPs of $\mathcal{CDA}$ relations [7, 8, 19], and the way to solve them [19]. Then come the definition of CSPs of $\mathcal{ROA}$ relations, and a discussion on how to adapt a known propagation algorithm [15, 16] to such CSPs. We finish the section with the presentation of CSPs combining both kinds of knowledge (CSPs of $\mathcal{CDA}$ relations and $\mathcal{ROA}$ relations on 2D points): most importantly, this last part will present in detail the propagation algorithm *PcS4c+()* we have already alluded to.

5.1 Reasoning within $\mathcal{CDA}$ and the $\mathcal{CDA}$-to-$\mathcal{ROA}$ interaction: the tables

The table in Figure 5 presents the augmented $\mathcal{CDA}$ composition table; for each pair (r_1, r_2) of $\mathcal{CDA}$ atoms, the table provides:

1. the standard composition, $r_1 \circ r_2$, of r_1 and r_2 [7, 8, 19]; and

$\frac{\circ}{\otimes}$	No	So	Ea	We	NE	NW	SE	SW
No	No	$[So, No]$	NE	NW	NE	NW	$[SE, NE]$	$[SW, NW]$
	br	$\{bp, cp, bw\}$	rr	lr	rr	lr	rr	lr
So	$[So, No]$	So	SE	SW	$[SE, NE]$	$[SW, NW]$	SE	SW
	$\{bp, cp, bw\}$	br	lr	rr	lr	rr	lr	rr
Ea	NE	SE	Ea	$[We, Ea]$	NE	$[NW, NE]$	SE	$[SW, SE]$
	lr	rr	br	$\{bp, cp, bw\}$	lr	lr	rr	rr
We	NW	SW	$[We, Ea]$	We	$[NW, NE]$	NW	$[SW, SE]$	SW
	rr	lr	$\{bp, cp, bw\}$	br	rr	rr	lr	lr
NE	NE	$[SE, NE]$	NE	$[NW, NE]$	NE	$[NW, NE]$	$[SE, NE]$	?
	lr	rr	rr	lr	$\{lr, br, rr\}$	lr	rr	$\{lr, bp, cp, bw, rr\}$
NW	NW	$[SW, NW]$	$[NW, NE]$	NW	$[NW, NE]$	NW	?	$[SW, NW]$
	rr	lr	rr	lr	rr	$\{lr, br, rr\}$	$\{lr, bp, cp, bw, rr\}$	lr
SE	$[SE, NE]$	SE	SE	$[SW, NE]$	$[SE, NE]$	?	SE	$[SW, NE]$
	lr	rr	lr	rr	lr	$\{lr, bp, cp, bw, rr\}$	$\{lr, br, rr\}$	rr
SW	$[SW, NW]$	SW	$[SW, SE]$	SW	?	$[SW, NW]$	$[SW, SE]$	SW
	rr	lr	lr	rr	$\{lr, bp, cp, bw, rr\}$	rr	lr	$\{lr, br, rr\}$

Fig. 5. The augmented composition table of the cardinal directions calculus: for each pair (r_1, r_2) of $\mathcal{CDA}$ atoms, the table provides the composition, $r_1 \circ r_2$, of r_1 and r_2, as well as the most specific $\mathcal{ROA}$ relation $r_1 \otimes r_2$ such that, for all 2D points x, y, z, the conjunction $r_1(x, y) \wedge r_2(y, z)$ logically implies $(r_1 \otimes r_2)(x, y, z)$. The question mark symbol ? represents the $\mathcal{CDA}$ universal relation $\{No, NW, We, SW, So, SE, Ea, NE, Eq\}$.

2. the most specific $\mathcal{ROA}$ relation $r_1 \otimes r_2$ such that, for all 2D points x, y, z, the conjunction $r_1(x, y) \wedge r_2(y, z)$ logically implies $(r_1 \otimes r_2)(x, y, z)$.

The operation $\circ$ is just the normal composition: it is internal to $\mathcal{CDA}$, in the sense that it takes as input two $\mathcal{CDA}$ atoms, and outputs a $\mathcal{CDA}$ relation. The operation $\otimes$, however, is not internal to $\mathcal{CDA}$, in the sense that it takes as input two $\mathcal{CDA}$ atoms, but outputs an $\mathcal{ROA}$ relation; $\otimes$ captures the interaction between $\mathcal{CDA}$ knowledge and $\mathcal{ROA}$ knowledge, in the direction $\mathcal{CDA}$-to-$\mathcal{ROA}$, by inferring $\mathcal{ROA}$ knowledge from given $\mathcal{CDA}$ knowledge. As an example for the new operation $\otimes$, from

$$SE(Berlin, London) \wedge No(London, Paris),$$

saying that Berlin is south-east of London, and that London is north of Paris, we infer the $\mathcal{ROA}$ relation lr on the triple $(Berlin, London, Paris)$:

$$lr(Berlin, London, Paris),$$

saying that, viewed from Berlin, Paris is to the left of London. As another example, from

$$No(Paris, Rome) \wedge So(Rome, London),$$

the most specific $\mathcal{ROA}$ relation we can infer on the triple $(Paris, Rome, London)$ is $\{bp, cp, bw\}$:

$$\{bp, cp, bw\}(Paris, Rome, London).$$

The reader is referred to [7, 8, 19] for the $\mathcal{CDA}$ converse table, providing the converse $r^{\smile}$ for each $\mathcal{CDA}$ atom r.

5.2 Reasoning within $\mathcal{ROA}$ and the $\mathcal{ROA}$-to-$\mathcal{CDA}$ interaction: the tables

Figure 6 provides for each of the $\mathcal{ROA}$ atoms, say t, the converse $t^{\smile}$ and the rotation $t^{\frown}$ of t. Figure 7 provides the $\mathcal{ROA}$ composition tables, which are computed in the

Region R	0	1	2	3	4	5	6	7	8
Atom t	de	dd	lr	bp	cp	bw	cr	br	rr
$t^{\smile}$	de	cp	rr	bp	dd	br	cr	bw	lr
$t^{\frown}$	de	cp	lr	bw	cr	br	dd	bp	rr

Fig. 6. For each of the nine regions $0, \ldots, 8$ in Figure 4(a-c), the corresponding $\mathcal{ROA}$ atom t, as well as the converse $t^{\smile}$ and the rotation $t^{\frown}$ of t.

following way. Given four 2D points x, y, z, w and two $\mathcal{ROA}$ atoms t_1 and t_2, the conjunction $t_1(x,y,z) \wedge t_2(x,z,w)$ is inconsistent if the most specific relation $b_1(x,z)$, one can infer from $t_1(x,y,z)$ on the pair (x,z), is different from the most specific relation $b_2(x,z)$, one can infer from $t_2(x,z,w)$ on the same pair (x,z). The $\mathcal{ROA}$ composition splits therefore into two composition tables, one for each of the following two cases:[2]

1. Case 1: $x = z$ (i.e., each of b_1 and b_2 is the relation $=$). This corresponds to $t_1 \in \{de, cp\}$ and $t_2 \in \{de, dd\}$.
2. Case 2: $x \neq z$ (i.e., each of b_1 and b_2 is the relation $\neq$). This corresponds to $t_1 \in \{dd, lr, bp, cp, bw, br, rr\}$ and $t_2 \in \{lr, bp, cp, bw, cr, br, rr\}$.

The $\mathcal{CDA}$ knowledge one can infer from $\mathcal{ROA}$ relations is presented in the table of Figure 8, which makes use of the following two functions, $\mathcal{L}ir$ and $\mathcal{R}ir$:

$$\mathcal{L}ir(r) = \begin{cases} \{SE, Ea, NE\} & \text{if } r = So, \\ \{SE, Ea, NE, No, NW\} & \text{if } r = SE, \\ \{NE, No, NW\} & \text{if } r = Ea, \\ \{NE, No, NW, We, SW\} & \text{if } r = NE, \\ \{NW, We, SW\} & \text{if } r = No, \\ \{NW, We, SW, So, SE\} & \text{if } r = NW, \\ \{SW, So, SE\} & \text{if } r = We, \\ \{SW, So, SE, Ea, NE\} & \text{if } r = SW. \end{cases} \quad \mathcal{R}ir(r) = \begin{cases} \{NW, We, SW\} & \text{if } r = So, \\ \{NW, We, SW, So, SE\} & \text{if } r = SE, \\ \{SW, So, SE\} & \text{if } r = Ea, \\ \{SW, So, SE, Ea, NE\} & \text{if } r = NE, \\ \{SE, Ea, NE\} & \text{if } r = No, \\ \{SE, Ea, NE, No, SW\} & \text{if } r = NW, \\ \{NE, No, NW\} & \text{if } r = We, \\ \{NE, No, NW, We, SW\} & \text{if } r = SW. \end{cases}$$

The function $\mathcal{L}ir$ ($\mathcal{L}$eft inferred relation) provides for its argument, say r (a $\mathcal{CDA}$ atom), the most specific $\mathcal{CDA}$ relation R such that for all x, y, z, the conjunction $r(x,y) \wedge lr(x,y,z)$ logically implies $R(x,z)$. For instance, if r is So then $R = \mathcal{L}ir(So) = \{SE, Ea, NE\}$ —from $So(Paris, London)$ and $lr(Paris, London, Madrid)$, we get $\{SE, Ea, NE\}(Paris, Madrid)$. As another example, if r is SE then $R = \mathcal{L}ir(SE) = \{SE, Ea, NE, No, NW\}$ —see the illustration of Figure 9: from $SE(Berlin, Hamburg)$ and $lr(Berlin, Hamburg, Paris)$, we get $\{SE, Ea, NE, No, NW\}(Berlin, Paris)$. The function $\mathcal{R}ir$ ($\mathcal{R}$ight inferred relation) is defined in a similar way, with lr replaced with rr.

Given a $c\mathcal{COA}$-CSP P, the table in Figure 8 illustrates how the $\mathcal{ROA}$ constraint $(\mathcal{T}^P)_{ijk}$ on the triple (X_i, X_j, X_k) of variables interacts with each of the three $\mathcal{CDA}$ constraints $(\mathcal{B}^P)_{ij}$, $(\mathcal{B}^P)_{ik}$ and $(\mathcal{B}^P)_{jk}$ on the pairs (X_i, X_j), (X_i, X_k) and (X_j, X_k). If $(\mathcal{T}^P)_{ijk}$ is an atomic relation, say r, then the interaction is given by the three functions roa-to-cda12, roa-to-cda13 and roa-to-cda23 of Figure 8; namely:

[2] A similar way of splitting the composition into more than one table has been followed for the ternary RA, $\mathcal{CYC}_t$, presented in [15, 16].

$\circ$	de	dd
de	de	dd
cp	cp	$\{lr, bp, bw, cr, br, rr\}$

$\circ$	lr	bp	cp	bw	cr	br	rr
dd	dd	dd	de	dd	dd	dd	dd
lr	$\{lr, bp, rr\}$	rr	cp	lr	lr	lr	$\{lr, bw, cr, br, rr\}$
bp	rr	$\{bw, cr, br\}$	cp	bp	bp	bp	lr
bw	lr	bp	cp	bw	bw	$\{bw, cr, br\}$	rr
cr	lr	bp	cp	bw	cr	br	rr
br	lr	bp	cp	$\{bw, cr, br\}$	br	br	rr
rr	$\{lr, bw, cr, br, rr\}$	lr	cp	rr	rr	rr	$\{lr, bp, rr\}$

Fig. 7. The $\mathcal{ROA}$ composition tables: in each of the two tables, the entry at the intersection of a line ℓ and a column c is the composition, $r_1 \circ r_2$, of r_1 and r_2, where r_1 is the $\mathcal{ROA}$ atom appearing as the first element of line ℓ and r_2 is the $\mathcal{ROA}$ atom appearing as the top element of column c.

1. $(\mathcal{B}^P)_{ij} \leftarrow \text{roa-to-cda12}(r, P, i, j, k)$;
   $(\mathcal{B}^P)_{ji} \leftarrow ((\mathcal{B}^P)_{ij})^{\smile}$;
2. $(\mathcal{B}^P)_{ik} \leftarrow \text{roa-to-cda13}(r, P, i, j, k)$;
   $(\mathcal{B}^P)_{ki} \leftarrow ((\mathcal{B}^P)_{ik})^{\smile}$;
3. $(\mathcal{B}^P)_{jk} \leftarrow \text{roa-to-cda23}(r, P, i, j, k)$;
   $(\mathcal{B}^P)_{kj} \leftarrow ((\mathcal{B}^P)_{jk})^{\smile}$;

If $(\mathcal{T}^P)_{ijk}$ is a disjunctive, non atomic relation, say R, then the interaction is the union of the interactions at the atomic level; namely:

1. $(\mathcal{B}^P)_{ij} \leftarrow \bigcup_{r \in R} \text{roa-to-cda12}(r, P, i, j, k)$;
   $(\mathcal{B}^P)_{ji} \leftarrow ((\mathcal{B}^P)_{ij})^{\smile}$;
2. $(\mathcal{B}^P)_{ik} \leftarrow \bigcup_{r \in R} \text{roa-to-cda13}(r, P, i, j, k)$;
   $(\mathcal{B}^P)_{ki} \leftarrow ((\mathcal{B}^P)_{ik})^{\smile}$;
3. $(\mathcal{B}^P)_{jk} \leftarrow \bigcup_{r \in R} \text{roa-to-cda23}(r, P, i, j, k)$;
   $(\mathcal{B}^P)_{kj} \leftarrow ((\mathcal{B}^P)_{jk})^{\smile}$;

5.3 CSPs of cardinal direction relations on 2D points

We define a $\mathcal{CDA}$-CSP as a CSP of which the constraints are $\mathcal{CDA}$ relations on pairs of the variables. The universe of a $\mathcal{CDA}$-CSP is the set $\mathbb{R}^2$ of 2D points.

A $\mathcal{CDA}$-matrix of order n is a binary constraint matrix of order n of which the entries are $\mathcal{CDA}$ relations. The constraint matrix associated with a $\mathcal{CDA}$-CSP is a $\mathcal{CDA}$-matrix.

A scenario of a $\mathcal{CDA}$-CSP is a refinement P' such that all entries of the constraint matrix of P' are atomic relations.

r	roa-to-cda12(r, P, i, j, k)	roa-to-cda13(r, P, i, j, k)	roa-to-cda23(r, P, i, j, k)
de	$(\mathcal{B}^P)_{ij} \cap \{Eq\}$	$(\mathcal{B}^P)_{ik} \cap \{Eq\}$	$(\mathcal{B}^P)_{jk} \cap \{Eq\}$
dd	$(\mathcal{B}^P)_{ij} \cap \{Eq\}$	$(\mathcal{B}^P)_{ik} \cap (\mathcal{B}^P)_{jk} \cap \overline{\{Eq\}}$	$(\mathcal{B}^P)_{ik}$
lr	$(\mathcal{B}^P)_{ij} \cap \overline{\{Eq\}}$	$(\mathcal{B}^P)_{ik} \cap \mathcal{L}ir((\mathcal{B}^P)_{ij})$	$(\mathcal{B}^P)_{jk} \cap \mathcal{R}ir((\mathcal{B}^P)_{ji})$
bp	$(\mathcal{B}^P)_{ij} \cap (\mathcal{B}^P)_{ki} \cap (\mathcal{B}^P)_{kj} \cap \overline{\{Eq\}}$	$(\mathcal{B}^P)_{ji}$	$(\mathcal{B}^P)_{ik}$
cp	$(\mathcal{B}^P)_{ij} \cap (\mathcal{B}^P)_{kj} \cap \overline{\{Eq\}}$	$(\mathcal{B}^P)_{ik} \cap \{Eq\}$	$(\mathcal{B}^P)_{ji}$
bw	$(\mathcal{B}^P)_{ij} \cap (\mathcal{B}^P)_{ik} \cap (\mathcal{B}^P)_{kj} \cap \overline{\{Eq\}}$	$(\mathcal{B}^P)_{ij}$	$(\mathcal{B}^P)_{ji}$
cr	$(\mathcal{B}^P)_{ij} \cap (\mathcal{B}^P)_{ik} \cap \overline{\{Eq\}}$	$(\mathcal{B}^P)_{ij}$	$(\mathcal{B}^P)_{jk} \cap \{Eq\}$
br	$(\mathcal{B}^P)_{ij} \cap (\mathcal{B}^P)_{ik} \cap (\mathcal{B}^P)_{jk} \cap \overline{\{Eq\}}$	$(\mathcal{B}^P)_{ij}$	$(\mathcal{B}^P)_{ij}$
rr	$(\mathcal{B}^P)_{ij} \cap \overline{\{Eq\}}$	$(\mathcal{B}^P)_{ik} \cap \mathcal{R}ir((\mathcal{B}^P)_{ij})$	$(\mathcal{B}^P)_{jk} \cap \mathcal{L}ir((\mathcal{B}^P)_{ji})$

Fig. 8. Given a $c\mathcal{COA}$-CSP P, the constraints imposed by the $\mathcal{ROA}$ relation $(\mathcal{T}^P)_{ijk}$ on the $\mathcal{CDA}$ relations on the different pairs of the three arguments. The table presents the case when $(\mathcal{T}^P)_{ijk}$ is an atomic relation, say r; the case when $(\mathcal{T}^P)_{ijk}$ is a disjunctive $\mathcal{ROA}$ relation is explained in the main text.

If we make the assumption that a $\mathcal{CDA}$-CSP does not include the empty constraint, which indicates a trivial inconsistency, then a $\mathcal{CDA}$-CSP is strongly 2-consistent.

Solving a $\mathcal{CDA}$-CSP

A simple adaptation of Allen's constraint propagation algorithm [1] can be used to achieve path consistency (hence strong 3-consistency) for $\mathcal{CDA}$-CSPs. Applied to a $\mathcal{CDA}$-CSP P, such an adaptation would repeat the following steps until either stability is reached or the empty relation is detected (indicating inconsistency):

1. Consider a triple (X_i, X_j, X_k) of variables verifying $(\mathcal{B}^P)_{ij} \not\subseteq (\mathcal{B}^P)_{ik} \circ (\mathcal{B}^P)_{kj}$
2. $(\mathcal{B}^P)_{ij} \leftarrow (\mathcal{B}^P)_{ij} \cap (\mathcal{B}^P)_{ik} \circ (\mathcal{B}^P)_{kj}$
3. If $((\mathcal{B}^P)_{ij} = \emptyset)$ then exit (the CSP is inconsistent).

Path consistency is complete for atomic $\mathcal{CDA}$-CSPs [19]. Given this, Ladkin and Reinefeld's solution search algorithm [18] can be used to search for a solution, if any, or otherwise report inconsistency, of a general $\mathcal{CDA}$-CSP.

5.4 CSPs on relative orientation of 2D points

We define an $\mathcal{ROA}$-CSP as a CSP of which the constraints are $\mathcal{ROA}$ relations on triples of the variables. The universe of an $\mathcal{ROA}$-CSP is the set $\mathbb{R}^2$ of 2D points.

An $\mathcal{ROA}$-matrix of order n is a ternary constraint matrix of order n of which the entries are $\mathcal{ROA}$ relations. The constraint matrix associated with an $\mathcal{ROA}$-CSP is an $\mathcal{ROA}$-matrix.

A scenario of an $\mathcal{ROA}$-CSP is a refinement P' such that all entries of the constraint matrix of P' are atomic relations.

If we make the assumption that an $\mathcal{ROA}$-CSP does not include the empty constraint, which indicates a trivial inconsistency, then an $\mathcal{ROA}$-CSP is strongly 3-consistent.

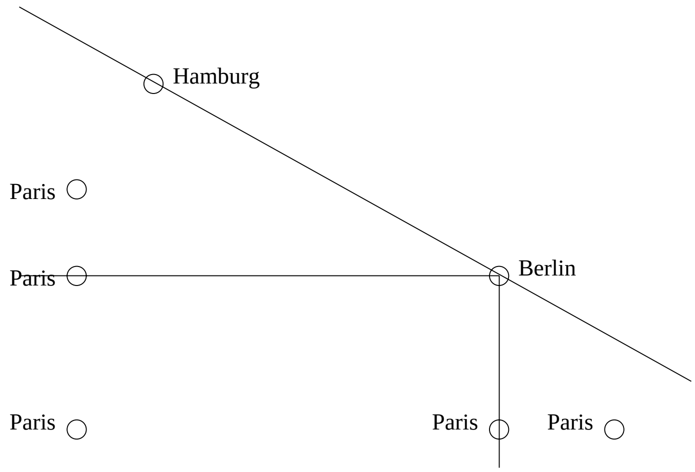

Fig. 9. From "Berlin is south-east of Hamburg" and "viewed from Berlin, Paris is to the left of Hamburg", we infer that "Berlin is south-east, east, north-east, north, or north-west of, Paris".

Searching for a strongly 4-consistent scenario of an $\mathcal{ROA}$-CSP

A simple adaptation of Isli and Cohn's constraint propagation algorithm [15, 16] can be used to achieve strong 4-consistency for $\mathcal{ROA}$-CSPs. Applied to an $\mathcal{ROA}$-CSP P, such an adaptation would repeat the following steps until either stability is reached or the empty relation is detected (indicating inconsistency):

1. Consider a quadruple (X_i, X_j, X_k, X_l) of variables verifying $(\mathcal{T}^P)_{ijl} \not\subseteq (\mathcal{T}^P)_{ijk} \circ (\mathcal{T}^P)_{ikl}$
2. $(\mathcal{T}^P)_{ijl} \leftarrow (\mathcal{T}^P)_{ijl} \cap (\mathcal{T}^P)_{ijk} \circ (\mathcal{B}^P)_{ikl}$
3. If $((\mathcal{T}^P)_{ijl} = \emptyset)$ then exit (the CSP is inconsistent).

Isli and Cohn have proposed a complete solution search algorithm for CSPs expressed in their $\mathcal{CYC}_t$ algebra [15, 16]. The algorithm is similar to the one of Ladkin and Reinefeld [18] for temporal interval networks [1], except that:

1. it refines the relation on a triple of variables at each node of the search tree, instead of the relation on a pair of variables; and
2. it makes use of a constraint propagation procedure achieving strong 4-consistency, in the preprocessing step and as the filtering method during the search, instead of a procedure achieving path consistency.

Unless we can prove that Isli and Cohn's strong 4-consistency procedure is complete for the $\mathcal{ROA}$ atomic relations, we cannot claim completeness of the solution search procedure for general $\mathcal{ROA}$-CSPs. But we can still use the procedure to search for a strongly 4-consistent scenario of the input CSP. For more details on the algorithm, and on its binary counterpart, the reader is referred to [15, 16, 18].

5.5 CSPs of cardinal directions relations and relative orientation relations on 2D points

We define a $c\mathcal{COA}$-CSP as a CSP of which the constraints consist of a conjunction of $\mathcal{CDA}$ relations on pairs of the variables, and $\mathcal{ROA}$ relations on triples of the variables. The universe of a $c\mathcal{COA}$-CSP is the set $\mathbb{R}^2$ of 2D points.

Matrix representation of a $c\mathcal{COA}$-CSP

A $c\mathcal{COA}$-CSP P can, in an obvious way, be represented as two constraint matrices:

1. a binary constraint matrix, $\mathcal{B}^P$, representing the $\mathcal{CDA}$ part of P, i.e., the subconjunction consisting of $\mathcal{CDA}$ relations on pairs of the variables; and
2. a ternary constraint matrix, $\mathcal{T}^P$, representing the $\mathcal{ROA}$ part of P, i.e., the rest of the conjunction, consisting of $\mathcal{ROA}$ relations on triples of the variables.

We refer to the representation as $\langle \mathcal{B}^P, \mathcal{T}^P \rangle$.

A constraint propagation procedure for $c\mathcal{COA}$-CSPs

A path consistency algorithm, such as Allen's [1], applied to a binary CSP such as a $\mathcal{CDA}$-CSP, uses a queue *Queue*, which can be supposed, for simplicity, to have been initialised to all pairs (x, y) of the CSP variables verifying $x \leq y$ (the variables are supposed to be ordered). The algorithm removes one pair of variables from *Queue* at a time; a removed pair is used to eventually update the relations on the neighbouring pairs of variables (pairs sharing at least one variable). Whenever such a pair is successfully updated, it is entered into *Queue*, if it is not already there, in order to be considered at a future stage for propagation. The algorithm terminates if the empty relation, indicating inconsistency, is detected, or if *Queue* becomes empty, indicating that a fixed point has been reached and the input CSP is made path consistent.

A strong 4-consistency algorithm, such as Isli and Cohn's [15, 16], applied to a ternary CSP such as an $\mathcal{ROA}$-CSP, is, somehow, an adaptation to ternary relations of a path consistency algorithm. It uses a queue *Queue*, which can be supposed, for simplicity, to have been initialised to all triples (x, y, z) of the CSP variables such that $x \leq y \leq z$. The algorithm removes one triple from *Queue* at a time; a removed triple is used to eventually update the relations on the neighbouring triples (sharing at least two variables). Whenever such a triple is successfully updated, it is entered into *Queue*, if it is not already there, in order to be considered at a future stage for propagation. The algorithm terminates if the empty relation, indicating inconsistency, is detected, or if *Queue* becomes empty, indicating that a fixed point has been reached and the input CSP is made strongly 4-consistent.

In Figure 10, we propose a constraint propagation procedure, *PcS4c+()*, for $c\mathcal{COA}$-CSPs, which aims at:

1. achieving path consistency (Pc) for the $\mathcal{CDA}$ projection;
2. achieving strong 4-consistency ($S4c$) for the $\mathcal{ROA}$ projection; and
3. more ($+$).

The procedure does more than just achieving path consistency for the $\mathcal{CDA}$ projection, and strong 4-consistency for the $\mathcal{ROA}$ projection. It implements as well the interaction between the two combined calculi; namely:

1. The path consistency operation, $(\mathcal{B}^P)_{ik} \leftarrow (\mathcal{B}^P)_{ik} \cap (\mathcal{B}^P)_{ij} \circ (\mathcal{B}^P)_{jk}$, which, under normal circumstances, operates internally, within a same CSP, should now be, and is, augmented so that it can send information from the $\mathcal{CDA}$ component into the $\mathcal{ROA}$ component; this is achieved by a call to the procedure *pair-propagation*(). Specifically, whenever a pair (X_i, X_j) of variables is taken from *Queue* for propagation, the following is performed for all variables X_k:
    - the procedure *pair-propagation*() of Figure 10 checks whether the relation on the pair (X_i, X_k) —see lines 1-4— or the relation on the pair (X_k, X_j) —see lines 6-9— can be successfully updated. If this happens, the corresponding pairs of variables are entered into *Queue* in order to be considered for propagation at a later point of the process. This part of the propagation is not new, and is widely known in the literature on propagation algorithms, such as path consistency (see [1] for Allen's well-known propagation algorithm, for the case of constraint-based qualitative temporal reasoning). What is new in the procedure *pair-propagation*() is the call to the procedure $\mathcal{CDA}$-to-$\mathcal{ROA}$() —see lines 5 and 10— which aims at checking, whenever a pair (X_i, X_j) is taken from *Queue*, whether the $\mathcal{CDA}$ relation on (X_i, X_j) can update the $\mathcal{ROA}$ relation on the triple (X_i, X_j, X_k) or that on the triple (X_k, X_i, X_j). If either of the two $\mathcal{ROA}$ relations gets successfully updated, the corresponding triple of variables is entered into *Queue* in order to be considered for propagation at a later point of the process. The procedure $\mathcal{CDA}$-to-$\mathcal{ROA}$() is the implementation of the $\mathcal{CDA}$-to-$\mathcal{ROA}$ interaction operation, $\otimes$, defined in the table of Figure 5, which outputs the $\mathcal{ROA}$ relation, $r \otimes s$, logically implied by the conjunction of two $\mathcal{CDA}$ atoms, r and s.
2. The strong 4-consistency operation, $(\mathcal{T}^P)_{ijk} \leftarrow (\mathcal{T}^P)_{ijk} \cap (\mathcal{T}^P)_{ijl} \circ (\mathcal{T}^P)_{ilk}$, which also operates internally under normal circumstances, is augmented so that it can send information from the $\mathcal{ROA}$ component into the $\mathcal{CDA}$ component; this is achieved by a call to the procedure *triple-propagation*(). Specifically, whenever a triple (X_i, X_j, X_k) is taken from *Queue* for propagation, the following is performed for all variables X_m:
    - the procedure *triple-propagation*() of Figure 10 checks whether the relation on the triple (X_i, X_j, X_m) —see lines 1-4— or the relation on the triple (X_i, X_k, X_m) —see lines 5-8— or the relation on the triple (X_j, X_k, X_m) —see lines 9-12— can be successfully updated. If this happens, the corresponding triples of variables are entered into *Queue* in order to be considered for propagation at a later point of the process. This part of the propagation is taken from Isli and Cohn's strong 4-consistency algorithm [15, 16]. What is new in the procedure *triple-propagation*() is the call to the procedure $\mathcal{ROA}$-to-$\mathcal{CDA}$() —see line 13— which aims at checking, whenever a triple (X_i, X_j, X_k) is taken from *Queue*, whether the $\mathcal{ROA}$ relation on (X_i, X_j, X_k) can update the $\mathcal{CDA}$ relations on the different pairs of the three arguments: the pairs (X_i, X_j), (X_i, X_k) and (X_j, X_k). If any of the three $\mathcal{CDA}$ relations gets successfully updated, the corresponding pair of variables is entered into *Queue* in order to be considered for propagation at a later point of the process. The procedure

$\mathcal{ROA}$-to-$\mathcal{CDA}$() is the implementation of the $\mathcal{ROA}$-to-$\mathcal{CDA}$ interaction table of Figure 8.

```
Input: the matrix representation ⟨B^P, T^P⟩ of a cCOA-CSP P with set of variables V.
Output: the CSP P made strongly 4-consistent.
procedure PcS4c+();
initialise Queue: Queue ← {(x, y) ∈ V^2 : x ≤ y} ∪ {(x, y, z) ∈ V^3 : x ≤ y ≤ z};
repeat{
  get (and remove) next element Q from Queue;
  if Q is a pair, say (X_i, X_j){
    for k ← 1 to n{pair-propagation(P, i, j, k);}
    }
  else (Q is a triple, say (X_i, X_j, X_k)){
    for m ← 1 to n{triple-propagation(P, i, j, k, m);}
    }
  }
until Queue is empty;

procedure pair-propagation(P, i, j, k);
        Temp ← (B^P)_ik ∩ (B^P)_ij ∘ (B^P)_jk;
        If Temp = ∅ then exit (the CSP is inconsistent);
        if Temp ≠ (B^P)_ik
            {add-to-queue(X_i, X_k); (B^P)_ik ← Temp; (B^P)_ki ← Temp⌣; }
        CDA-to-ROA(P, i, j, k);
        Temp ← (B^P)_kj ∩ (B^P)_ki ∘ (B^P)_ij;
        If Temp = ∅ then exit (the CSP is inconsistent);
        if Temp ≠ (B^P)_kj
            {add-to-queue(X_k, X_j); (B^P)_kj ← Temp; (B^P)_jk ← Temp⌣; }
        CDA-to-ROA(P, k, i, j);

procedure triple-propagation(P, i, j, k, m);
        Temp ← (T^P)_ijm ∩ (T^P)_ijk ∘ (T^P)_ikm;
        If Temp = ∅ then exit (the CSP is inconsistent);
        if Temp ≠ (T^P)_ijm
            {add-to-queue(X_i, X_j, X_m);update(P, i, j, m, Temp);}
        Temp ← (T^P)_ikm ∩ (T^P)_ikj ∘ (T^P)_ijm;
        If Temp = ∅ then exit (the CSP is inconsistent);
        if Temp ≠ (T^P)_ikm
            {add-to-queue(X_i, X_k, X_m);update(P, i, k, m, Temp);}
        Temp ← (T^P)_jkm ∩ (T^P)_jki ∘ (T^P)_jim;
        If Temp = ∅ then exit (the CSP is inconsistent);
        if Temp ≠ (T^P)_jkm
            {add-to-queue(X_j, X_k, X_m);update(P, j, k, m, Temp);}
        ROA-to-CDA(P, i, j, k);

procedure update(P, i, j, k, T);
(T^P)_ijk ← T; (T^P)_ikj ← T⌣; (T^P)_jki ← T⌢;
(T^P)_jik ← ((T^P)_jki)⌣; (T^P)_kij ← ((T^P)_jki)⌢; (T^P)_kji ← ((T^P)_kij)⌣;
```

Fig. 10. A constraint propagation procedure, *PcS4c+()*, for $c\mathcal{COA}$-CSPs. The procedures $\mathcal{CDA}$-to-$\mathcal{ROA}$ and $\mathcal{ROA}$-to-$\mathcal{CDA}$ used by the algorithm are defined in Figure 11.

Theorem 1 *The constraint propagation procedure* PcS4c+() *runs into completion in* $O(n^4)$ *time, where* n *is the number of variables of the input* $c\mathcal{COA}$*-CSP.*

Proof. The number of variable pairs is $O(n^2)$, whereas the number of variable triples is $O(n^3)$. A pair as well as a triple may be placed in Queue at most a constant number of times (9 for a pair, which is the total number of $\mathcal{CDA}$ atoms; and also 9 for a triple, *procedure* $\mathcal{CDA}$-to-$\mathcal{ROA}(P,i,j,k)$;

roa-ir $\leftarrow \bigcup_{r_1\in(\mathcal{B}^P)_{ij},\, r_2\in(\mathcal{B}^P)_{jk}} r_1 \otimes r_2$;
$Temp \leftarrow (\mathcal{T}^P)_{ijk} \cap$ roa-ir;
If $Temp = \emptyset$ then exit (the CSP is inconsistent);
if $Temp \neq (\mathcal{T}^P)_{ijk}$
  {add-to-queue(X_i, X_j, X_k);*update*$(P,i,j,k,Temp)$;}

procedure $\mathcal{ROA}$-to-$\mathcal{CDA}(P,i,j,k)$;

$Temp \leftarrow \bigcup_{r\in R}$ roa-to-cda12(r,P,i,j,k);
If $Temp = \emptyset$ then exit (the CSP is inconsistent);
if $Temp \neq (\mathcal{T}^P)_{ij}$
  {add-to-queue(X_i, X_j); $(\mathcal{B}^P)_{ij} \leftarrow Temp$; $(\mathcal{B}^P)_{ji} \leftarrow Temp^{\smile}$; }
$Temp \leftarrow \bigcup_{r\in R}$ roa-to-cda13(r,P,i,j,k);
If $Temp = \emptyset$ then exit (the CSP is inconsistent);
if $Temp \neq (\mathcal{T}^P)_{ik}$
  {add-to-queue(X_i, X_k); $(\mathcal{B}^P)_{ik} \leftarrow Temp$; $(\mathcal{B}^P)_{ki} \leftarrow Temp^{\smile}$; }
$Temp \leftarrow \bigcup_{r\in R}$ roa-to-cda23(r,P,i,j,k);
If $Temp = \emptyset$ then exit (the CSP is inconsistent);
if $Temp \neq (\mathcal{T}^P)_{jk}$
  {add-to-queue(X_j, X_k); $(\mathcal{B}^P)_{jk} \leftarrow Temp$; $(\mathcal{B}^P)_{kj} \leftarrow Temp^{\smile}$; }

Fig. 11. The procedures $\mathcal{CDA}$-to-$\mathcal{ROA}$ and $\mathcal{ROA}$-to-$\mathcal{CDA}$ used by the constraint propagation algorithm *PcS4c+()* of Figure 10.

which is the total number of $\mathcal{ROA}$ atoms). Every time a pair or a triple is removed from Queue for propagation, the procedure performs $O(n)$ operations. ■

Example 2 *Consider again the description of Example 1. We can represent the situation as a c$\mathcal{COA}$-CSP with variables X_b, X_h, X_l and X_p, standing for the cities of Berlin, Hamburg, London and Paris, respectively.*

1. *The knowledge "viewed from Hamburg, Berlin is to the left of Paris" translates into the $\mathcal{ROA}$ constraint* $\text{lr}(X_h, X_p, X_b)$*:* $(\mathcal{T}^P)_{hpb} = \{\text{lr}\}$.
2. *The other $\mathcal{ROA}$ knowledge translates as follows:* $(\mathcal{T}^P)_{hlp} = \{\text{lr}\}$, $(\mathcal{T}^P)_{hlb} = \{\text{lr}\}$, $(\mathcal{T}^P)_{lpb} = \{\text{lr}\}$.
3. *The $\mathcal{CDA}$ part of the knowledge translates as follows:* $(\mathcal{B}^P)_{hp} = \{\text{No}\}$, $(\mathcal{B}^P)_{hb} = \{\text{NW}\}$, $(\mathcal{B}^P)_{pl} = \{\text{So}\}$.

As discussed in Example 1, reasoning separately about the two components of the knowledge shows two consistent components, whereas the combined knowledge is clearly inconsistent. Using the procedure PcS4c+()*, we can detect the inconsistency in the following way. From the $\mathcal{CDA}$ constraints* $(\mathcal{B}^P)_{hp} = \{\text{No}\}$ *and* $(\mathcal{B}^P)_{pl} = \{\text{So}\}$*, the algorithm infers, using the augmented $\mathcal{CDA}$ composition table of Figure 5 —specificaly, the $\mathcal{CDA}$-to-$\mathcal{ROA}$ interaction operation $\otimes$— the $\mathcal{ROA}$ relation* $\{\text{bp}, \text{cp}, \text{bw}\}$ *on the triple (X_h, X_p, X_l). The conjunction of the inferred knowledge* $\{\text{bp}, \text{cp}, \text{bw}\}(X_h, X_p, X_l)$ *and the already existing knowledge* $\{\text{lr}\}(X_h, X_l, X_p)$ *—equivalent to* $\{\text{rr}\}(X_h, X_p, X_l)$*— gives the empty relation, indicating the inconsistency of the knowledge.*

6 Summary

We have presented the combination of two calculi of spatial relations well-known in Qualitative Spatial Reasoning (QSR): Frank's cardinal direction calculus [7, 8] and Freksa's relative orientation calculus [9, 10]. With an example illustrating the importance of such a combination to Geographical Information Systems (*GIS*), we have shown that reducing the issue of reasoning about knowledge expressed in the combined language to a simple matter of reasoning separately about each of the two components was not sufficient. The interaction between the two kinds of knowledge has thus to be handled: we have provided a constraint propagation algorithm for such a purpose, which:

1. achieves path consistency for the cardinal direction component;
2. achieves strong 4-consistency for the relative orientation component; and
3. implements the interaction between the two kinds of knowledge.

Combining and integrating different kinds of knowledge is an emerging and challenging issue in QSR. Related work has been done by Gerevini and Renz [13], which deals with the combination of topological knowledge and relative size knowledge in QSR. Similar work might be carried out for other aspects of knowledge in QSR, such as qualitative distance [2] and relative orientation [9, 10], a combination known to be highly important for *GIS* and robot navigation applications, and on which not much has been achieved so far.

This work has been carried out within the context of a DFG project "Description Logics and Spatial reasoning" (DLS). The goal of the project is to use description logics with concrete domains, where the concrete domain is a language of spatial relations. Work on description logics with a language of qualitative spatial relations as a concrete domain can be found in the literature, such as the one in [14]; work that has been, however, restricted to the case of binary relations, such as the RCC-8 relations in [14]. The definition of qualitative spatial languages of ternary relations is an emerging and promising issue in QSR [9, 10, 15, 16], and one of the main goals of our project is to define description logics with concrete domains, where the concrete domain is a language of spatial ternary relations, or a language of spatial ternary relations and spatial binary relations, as the one we have defined in this work.

References

1. J F Allen. Maintaining knowledge about temporal intervals. *Communications of the Association for Computing Machinery*, 26(11):832–843, 1983.
2. E Clementini, P Di Felice, and D Hernández. Qualitative representation of positional information. *Artificial Intelligence*, 95:317–356, 1997.
3. A G Cohn. Qualitative spatial representation and reasoning techniques. In *Proceedings KI: German Annual Conference on Artificial Intelligence*, volume 1303 of *Lecture Notes in Artificial Intelligence*, pages 1–30. Springer-Verlag, 1997.
4. A De Morgan. On the syllogism, no. iv, and on the logic of relations. *Trans. Cambridge Philos. Soc. 10*, pages 331–358, 1864.
5. A De Morgan. *On the Syllogism and other Logical Writings.* Yale University Press, New Haven, 1966.
6. R Dechter. From local to global consistency. *Artificial Intelligence*, 55:87–107, 1992.
7. A U Frank. Qualitative spatial reasoning with cardinal directions. In *Proceedings of the Austrian Conference on Artificial Intelligence*, pages 157–167, Vienna, 1991.

8. A U Frank. Qualitative spatial reasoning about distances and directions in geographic space. *Journal of Visual Languages and Computing*, 3:343–371, 1992.
9. C Freksa. Using orientation information for qualitative spatial reasoning. In A U Frank, I Campari, and U Formentini, editors, *Proceedings of International Conference on Theories and Methods of Spatio-Temporal Reasoning in Geographic Space*, Berlin, 1992. Springer-verlag.
10. C Freksa and K Zimmermann. On the utilization of spatial structures for cognitively plausible and efficient reasoning. In *Proceedings of the 1992 IEEE International Conference on Systems, Man and Cybernetics*, 1992.
11. E C Freuder. Synthesizing constraint expressions. *Communications of the Association for Computing Machinery*, 21:958–966, 1978.
12. E C Freuder. A sufficient condition for backtrack-free search. *Journal of the Association for Computing Machinery*, 29:24–32, 1982.
13. A Gerevini and J Renz. Combining topological and qualitative size constraints for spatial reasoning. In *Proceedings of the Fourth International Conference on Principles and Practice of Constraint Programming (CP)*, Berlin, 1998.
14. V Haarslev, C Lutz, and R Möller. A description logic with concrete domains and a role-forming predicate operator. *Journal of Logic and Computation*, 9(3):351–384, 1999.
15. A Isli and A G Cohn. An algebra for cyclic ordering of 2D orientations. In *Proceedings AAAI*, pages 643–649, Madison, WI, 1998. AAAI Press.
16. 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.
17. P Ladkin and R Maddux. On binary constraint problems. *Journal of the Association for Computing Machinery*, 41(3):435–469, 1994.
18. P Ladkin and A Reinefeld. Effective solution of qualitative constraint problems. *Artificial Intelligence*, 57:105–124, 1992.
19. G Ligozat. Reasoning about cardinal directions. *Journal of Visual Languages and Computing*, 9(1):23–44, 1998.
20. A K Mackworth. Consistency in networks of relations. *Artificial Intelligence*, 8:99–118, 1977.
21. U Montanari. Networks of constraints: Fundamental properties and applications to picture processing. *Information Sciences*, 7:95–132, 1974.
22. A Tarski. On the calculus of relations. *Journal of Symbolic Logic*, 6:73–89, 1941.
23. K Zimmermann and C Freksa. Qualitative spatial reasoning using orientation, distance, and path knowledge. *Applied Intelligence*, 6:49–58, 1996.