# Dependency Calculus: Reasoning in a General Point Relation Algebra

Marco Ragni and Alexander Scivos

Institut für Informatik, Albert-Ludwigs-Universität Freiburg,
Georges-Köhler-Allee 52, 79110 Freiburg, Germany
{ragni, scivos}@informatik.uni-freiburg.de

**Abstract.** Reasoning about complex dependencies between events is a crucial task. However, qualitative reasoning has so far concentrated on spatial and temporal issues. In contrast, we present a new dependency calculus ($DC$) that is created for specific questions of reasoning about causal relations and consequences. Applications in the field of spatial representation and reasoning are, for instance, modeling traffic networks, ecological systems, medical diagnostics, and Bayesian Networks. Several extensions of the fundamental linear point algebra have been investigated, for instance on trees or on nonlinear structures. $DC$ is an improved generalization that meets all requirements to describe dependencies on networks. We investigate this structure with respect to satisfiability problems, construction problems, tractable subclassses, and embeddings into other relation algebras. Finally, we analyze the associated interval algebra on network structures.

## 1 Introduction

Reasoning about complex dependencies between events is a crucial task in many applications when decisions need to be made. Whenever the required answer is a decision or classification, Qualitative Reasoning (QR) is best-suited: It abstracts from metrical details of the physical world and enables computers to make predictions about relations, even when precise quantitative information is not available or irrelevant [5]. QR is an abstraction that summarizes similar quantitative states into one qualitative characterization. From the cognitive perspective, the qualitative method *categorizes* features within the object domain rather than by *measuring* them in terms of some external scale [7]. This is the reason why qualitative descriptions are quite natural for humans.

The two main directions in QR so far are spatial and temporal reasoning. In terms of spatial reasoning, topological reasoning about regions [11], positional reasoning about point configurations [7], and reasoning about directions can be distinguished. For temporal reasoning, either points [14] or intervals [1] are used as basic entities.

In contrast, we present a calculus that is created for a new direction in QR: specific questions of reasoning about causal relations and consequences. Our approach is purely qualitative, hence in comparison with Bayesian networks, no time for computing the probabilities is needed. Nevertheless, dependencies can reliably be derived. We show how known relation algebras can be refined to achieve this goal.

The linear point algebra $PA_{lin}$ introduced by Vilain [14] is one of the most prominent and fundamental formalisms in the domain of qualitative spatial and temporal rea-

U. Furbach (Ed.): KI 2005, LNAI 3698, pp. 49–63, 2005.

soning. Its basic relations are $\{\prec, =, \succ\}$. Many widely used algebras, like Allen's interval algebra [1] or the Cardinal Directions calculus, can be constructed by $PA_{lin}$ relations. The general satisfiability problem for $PA_{lin}$ is in P [14], but for the associated interval algebra $IA_{lin}$, it is NP-complete. NP-hard problems usually have interesting fragments - so called tractable subclasses. A subclass of a relational algebra is a subset of relations closed under composition and converse. A subclass is called tractable if satisfiability can be decided in polynomial time. Normally, for these classes the path-consistency method decides satisfiability. Nebel et al. [10] identified all maximal tractable subclasses for $IA_{lin}$.

However, real-world problems have not necessarily linear structures as underlying space. Starting in the nineties, there have been extensions of the linear case into treelike and nonlinear structures [2], [3] to address such problems. The point algebra $PA_{br}$ for treelike, often called branching structures, consists of the basic relations $\{\prec, =, \succ, \|\}$, whereby $\|$ states that two points are on different branches. Contrary to the point algebra of linear time, satisfiability testing for branching time is NP-hard. Moreover, the associated satisfiability problem for $PA_{br}$ is NP-complete. Broxvall [3] identified five maximal tractable subsets of $PA_{br}$. The most general known point algebra, up to now, consists of the basic relations $\{\prec, =, \succ, \|\}$ which is the point algebra on nonlinear structures $PA_{po}$. Broxvall showed that the satisfiability problem is in the same class as for branching structures and identified three maximal tractable fragments.

But, these approaches are still too coarse for some applications. For instance, to identify dependencies, it is not only important to state that two points are unrelated, but also to qualify if two points or states have a common ancestor or if there is no such 'decision' point. For this reason, we define a relation algebra qualifying decision points and get a proper generalization of Broxvall's nonlinear relation algebra [3]. This new algebra, called dependency calculus ($DC$), meets all requirements to describe dependencies in networks. This is demonstrated by applications in the field of spatial representation and reasoning. There are two aspects: dependencies of points are described by the point algebra $PA_{dc}$, and based on this we define the associated interval algebra $IA_{dc}$.

Analyzing the composition tables shows that the dependency calculus for point structures $PA_{dc}$ is a proper generalization of $PA_{po}$, which leads to a more precise reasoning. For that reason, it would be also useful to use the new algebra $PA_{dc}$ instead of using $PA_{po}$ in nonlinear structures, if $PA_{dc}$ is still in the same complexity class. This shows the need to analyze the complexity of the depencency calculus.

One main concept is that it is not only important to analyze questions concerning satisfiability problems, but also to show the correspondence between $DC$ and other relation algebras. To be more precise: Is there a relation algebra which is so similar to this dependency calculus that there is an isomorphism between these two algebras? An isomorphism with the property to preserve the tractability of subclasses? The method of identifying isomorphisms, or at least homomorphisms between calculi seems to be a promising idea to structure the field of relation algebras by the mathematical concept of homomorphism. Such mappings transfer algebraic aspects and complexity results from one algebra to another.

This paper is organized as follows: In Section 2, we present application areas and pose some important questions. In Section 3, we review the theory of partial orders and

sketch known results concerning the point algebra on different structures. We present the set $PA_{dc}$ of extensions of basic relations in our general framework. Then, the concept of correspondence of relation algebras is outlined. In a next step, we investigate the computational complexity of constraint satisfaction problems of $PA_{dc}$. In particular, we show that the satisfiability problem is NP-complete and all maximal tractable subclasses of $PA_{dc}$ are presented by an algebraic correspondence to RCC-5. In Section 4, we introduce the associated interval algebra $IA_{dc}$ on directed acyclic graphs, which allows to distinguish more constellations than classical interval algebras (even on nonlinear structures). We identify tractable subclasses and prove NP-completeness of the satisfiability problem for $IA_{dc}$. Finally, Section 5 summarizes our results and raises questions that are left open.

## 2 Application Areas

Various tasks in management and science require a good understanding of complex dependencies, for instance monitoring the distribution of pollution in ecological networks, identifying delivery bottlenecks in a supply chain network, inhibiting the spreading of deseases, or minimizing delays in a railway system.

For instance, if we observe pollution in an ecosystem of flowing water, we can draw conclusions about pollution at other points. Regard the flow network shown in Fig. 1. In

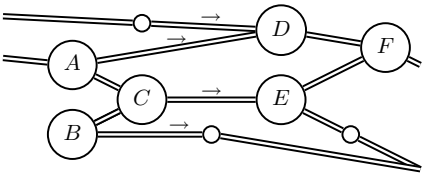


**Fig. 1.** A pipe network. Flow occurs along the "pipes" from left to right. Is there a difference between the pairs $(A, B)$ and $(D, E)$?

this example, if pollution is found at point $D$, point $F$ will be polluted as well. It might be caused from a source at point $A$. Points $B$, $C$, and $E$ could not cause the pollution at point $D$. Nevertheless, there is a connection between $C$ and $D$: Knowing that $C$ is polluted, it is likely that $D$ is polluted by the same substance (and vice versa). Does the same hold true for $B$ and $D$? It does not because they have no common point upstream. We present a calculus that directly represents such differences. Therefore, it is vital to have a new relation $\curlywedge$ which we call "fork" for pairs like $(C, D)$ or $(D, E)$ that share a common ancestor. The set of basic relations "fork" ($\curlywedge$), "before" ($\prec$), "equal" ($=$), and "after" ($\succ$), we call "dependent". The only other case, like $(A, B)$, we call "independent" ($\asymp$). If not all dependencies are known, some relations are not completely specified. Such imprecise knowledge is described by unions of these basic relations.

Dependencies of probabilities (when observations depend on each other, like in the water flow example) are often described by a Bayesian network. The answer whether two random variables are independent is based on the structure of the network. Can we

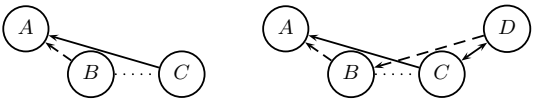


**Fig. 2.** A virus transmittance scheme. Arrows indicate assured, dashed ones unassured donorship, dotted lines mean that both persons carry the same virus. No lines means that we do not have prior knowledge. The situation on the left must be incomplete. The existence of a fourth person $D$ accounts for this. $PA_{dc}$ concludes that indeed there was indirect transmittance from $D$ to $A$.

automatically deduce from a given set of direct dependencies, which pairs are "independent" and which are "dependent"? For instance, in Fig. 1, observations at $A$ and $B$ are independent, but $E$ and $D$ are not.

Another important type of task is network design, that is finding a feasible partial order for a given system of nodes under certain order contstraints. For example, in a supply network, constraints could be that a warehouse has (indirectly) to be delivered from a specific production site, and that two warehouses do not have a common "bottleneck" site delivering both. Other examples are train scheduling (which connecting trains should wait for a delayed train, which not, to avoid critical dependencies?), automated planning, project organization and program verification. The task is to find a cycle-free order of the procedures such that one procedure delivers its result before the next routine is started. How time-consuming is the task of finding such an order?

In all these cases, the order may remain partly unspecified between elements for which the order does not matter. In other cases, the specific order (dependency) is vital, and it is necessary to deduce the hidden truth. For example, when tracking the spreading of a contagious disease, it is not always clear who was the donor, or if there was contact at all. Often, reasoning based on the known dependencies restricts the possibilities in cases of uncertainty or helps to detect formerly unknown causes (as in the example given in Fig. 2). These examples show that reasoning about dependencies, even with uncertainty, is highly important. In all these problems it matters if two nodes in the dependency graph have a common ancestor or not. In order to do automated reasoning, a calculus is needed that expresses this relation between a pair of nodes. This helps to distinguish between possibly affected spots and those that are unaffected. In which cases do efficient algorithms exist to detect causes and implications, to discover dependencies, and to find an order compliant with a given specification?

We present and investigate a calculus that suits to represent and deduce knowledge about dependencies by extending the language of partial ordering. This calculus is useful in various applications dealing with reasoning about spatial, temporal, spatio-temporal, topological, competitive, or causal relations.

## 3 The Dependency Calculus and Partial Orders

### 3.1 Formal Definition of $PA_{dc}$

To begin with the formalization of these concepts, let us recall the definition of a partial order: A partial order is a relation $\leq \subset A \times A$ that satisfies the following three properties for any $a, b, c \in A$:

i. Reflexivity: $a \leq a$.
ii. Antisymmetry: If $a \leq b$ and $b \leq a$, then $a = b$.
iii. Transitivity: If $a \leq b$ and $b \leq c$, then $a \leq c$.

A total order is a partial order that satisfies a fourth property:

iv. Comparability: For any $a, b \in A$, either $a \leq b$ or $b \leq a$.

$PA_{dc}$ is based on the notion of relations between pairs of variables interpreted as elements of a partial order. We consider five basic relations, which we denote by $\prec, =, \succ, \curlywedge, \asymp$. If $x, y$ are points in a partial order $\langle T, \leq \rangle$, then we define these relations in terms of the partial order as follows:

$x \prec y$ iff $x \leq y$ and not $y \leq x$.
$x = y$ iff $x \leq y$ and $y \leq x$.
$x \succ y$ iff $y \leq x$ and not $x \leq y$.
$x \curlywedge y$ iff $\exists z\ z \leq y \wedge z \leq x$ and neither $x \leq y$ nor $y \leq x$.
$x \asymp y$ iff neither $\exists z\ z \leq y \wedge z \leq x$ nor $x \leq y$ nor $y \leq x$ .

Semantically, a constraint $v_1\ R\ v_2$ holds in a partial order $(T, \leq)$ that complies with this definition. All relations between nodes in Fig. 1 can be described by these five basic relations. For describing trees, the four relations $\{\prec, =, \succ, \curlywedge\}$ are sufficient. Therefore, whenever the relation $\asymp$ occurs, the graph cannot be a tree.

A *boolean algebra* contains all unions $\cup$, intersections $\cap$, and complements $^-$ of a set of basic relations. If a boolean algebra is closed under additional operations composition $\circ$ and inverse $^{-1}$, it is called *relation algebra*, whereby

$$x \quad R^{-1} \quad y \quad \text{holds iff } y\ R\ x,$$
$$x\ (R_1 \circ R_2)\ y \text{ holds iff } \exists z \quad x\ R_1\ z \text{ and } z\ R_2\ y.$$

$PA_{dc}$ is the concrete binary finite relation algebra [8] generated by $\{\prec, =, \succ, \curlywedge, \asymp\}$. Furthermore, given an atomic relation algebra $A$ with finite atom set $B(A)$ (i.e. $B(A)$ is the set of all basic relations), each relation $r \in A$ can be written in a unique manner as a union of basic relations $b_1, \ldots, b_n$. Algebraic functions such as composition, converse, intersection, union, and complement, can be computed from basic relations:

$$(b_1 \cup \cdots \cup b_t) \circ (b'_1 \cup \cdots \cup b'_l) = \bigcup_{1 \leq i \leq t, 1 \leq j \leq l} (b_i \circ b'_j) \tag{1}$$

$$(b_1 \cup \cdots \cup b_l)^{-1} = (b_1^{-1} \cup \cdots \cup b_l^{-1}) \tag{2}$$

Hence, it is sufficient to analyze the basic relations. For $B(PA_{dc})$, the composition operator is defined in Table 1. Composition tables are important for constraint based reasoning. For instance, the path-consistency algorithm [9], which can be used to identify some inconsistent networks, uses the composition table. $PA_{dc}$ is the relation algebra generated by the 5 basic relations $\prec, =, \succ, \curlywedge, \asymp$ and the associated composition as shown by Table 1. The universal relation $\{\prec, =, \succ, \asymp, \curlywedge\}$ is denoted by $\top$.

**Table 1.** The composition table of $PA_{dc}$

| $\circ$ | $\prec$ | $=$ | $\succ$ | $\asymp$ | $\perp$ |
|---|---|---|---|---|---|
| $\prec$ | $\prec$ | $\prec$ | $\top$ | $\asymp$ | $\prec, \perp, \asymp$ |
| $=$ | $\prec$ | $=$ | $\succ$ | $\asymp$ | $\perp$ |
| $\succ$ | $\prec, =, \succ, \perp$ | $\succ$ | $\succ$ | $\succ, \asymp, \perp$ | $\succ, \perp$ |
| $\asymp$ | $\prec, \asymp, \perp$ | $\asymp$ | $\asymp$ | $\top$ | $\prec, \asymp, \perp$ |
| $\perp$ | $\prec, \perp$ | $\perp$ | $\succ, \asymp, \perp$ | $\succ, \asymp, \perp$ | $\top$ |

### 3.2 Computational Complexity

Assume that a set of constraints between some points is given. One question might be whether this set is consistent: Is it possible to construct a network in which all the constraints are satisfied? And, what is the computational effort for constructing it?

**Definition 1.** *Let $\mathcal{R} \subseteq PA_{dc}$ be a set of point relations and $\mathcal{C}$ a class of partial orders. A constraint system is a directed multigraph $\Pi = (V, E)$, where the nodes in $V$ are point variables and $E \subseteq V \times \mathcal{R} \times V$ denotes the constraints imposed on the variables. A tuple $(f, (T, \leq))$ where $f : V \to T$ is a total function and $(T, \leq) \in \mathcal{C}$ is called an interpretation of $\Pi$. It is satisfiable iff there exists an interpretation $\mathcal{M} = (f, (T, \leq))$ such that $f(u)\ R\ f(v)$ holds for every $(u, R, v) \in E$. $\mathcal{M}$ is called a model of $\Pi$. There are two types of problems for such a given constraint system:*

*i. The satisfiability problem $SAT_{\mathcal{C}}(\mathcal{R})$: Is $\Pi$ satisfiable?*
*ii. The network design problem $NDP_{\mathcal{C}}(\mathcal{R})$: Find a model of $\Pi$.*

*The size of a problem instance $(V, E)$ is $|V| + |E|$.*

In an interpretation, a partial order is chosen such that each variable $V$ is assigned to an element of the partial order. But this assignment is not required to be surjective: Not all elements of the partial order must correspond to a variable.

This definition of a satisfiability problem is a generalization of the classical definition of a *Constraint Satisfaction Problem* (**CSP**): Given a description consisting of

- a set $V$ of $n$ variables $\{v_1, ..., v_n\}$,
- the possible values $D_i$ of variables $v_i$,
- constraints (sets of relations) over subsets of variables,

Is it possible to find an assignment $v_i \mapsto D_i$ satisfying all constraints?

In our case, the possible values $D_i$ are not fixed. We only know that the possible values $D_i$ are part of a member in the class of partial orders $\mathcal{P}$. A CSP is a satisfiability problem with a class that consists just of one given partial order $(T, \leq)$.

**Lemma 1.** *For a class $\mathcal{C}$ of partial orders and set of relations $\mathcal{R}$,*
*a. if $SAT_{\mathcal{C}}(\mathcal{R})$ is in* P*, the same holds for $NDP_{\mathcal{C}}(\mathcal{R})$, and vice versa.*
*b. if $SAT_{\mathcal{C}}(\mathcal{R})$ is* NP*-complete, $NDP_{\mathcal{C}}(\mathcal{R})$ is* NP*-complete, too.*

*Proof.* Obviously, $NDP_{\mathcal{C}}(\mathcal{R})$ is at least as difficult as $SAT_{\mathcal{C}}(\mathcal{R})$.
For a., assume that $SAT_{\mathcal{C}}(\mathcal{R})$ is in P. A PTIME algorithm for $NDP_{\mathcal{C}}(\mathcal{R})$ is given by:

```
for c = (x1, R, x2) ∈ E:                          \\ choose a constraint
    set possible [c] ← false;
    for b ∈ R:                                    \\ choose a basic relation
        set c ← (x1, {b}, x2);                    \\ fix basic relation b
        if satisfiable (new constraint system):   \\ test is in PTIME, by assumption
            set possible [c] ← true; exit inner loop;
    endfor;                                       \\ try next basic relation
    if possible [c] = false:
        return NOT SATISFIABLE;                   \\ no feasible basic relation found
endfor;                                           \\ take next constraint
return SATISFIABLE.                               \\ as all constraints are possible
```

For b., a scenario can be guessed, i.e. a jointly satisfiable set of basic relations. Therefore, by assumption, a model can be guessed and verified in polynomial time.

It is easy to transform a directed acyclic graph into a partial order by taking the transitive and reflexive closure. Let $\mathcal{DAG}$ be the class of partial orders obtained by such transformations. What can we expect in terms of complexity for $SAT_{\mathcal{DAG}}(PA_{dc})$? We call this the "general" case and write $SAT(PA_{dc})$.

An interesting question is whether the relation $\asymp$ in $PA_{dc}$ provides additional complexity in comparison with $PA_{po}$. Or is the general satisfiability problem still NP-complete (cf. [14])? Assume that we have two relation algebras given. One of these relation algebras has been perfectly analyzed in terms of complexity questions and tractable fragments. Can we find a mapping from one algebra to the other, preserving all relevant properties, to transfer all results? By the following definitions, we introduce the concept of homomorphisms and isomorphisms between relation algebras, and we will extend this to a certain kind of reduction - a reduction which preserves the tractability.

**Definition 2.** *For relation algebras $\Gamma, \Gamma'$ a homomorphism is a function $\gamma$ from $\Gamma$ to $\Gamma'$ such that $\gamma$ preserves all operations of the boolean algebra and for relations $R$, $S$:*

1. $\gamma(R^{-1}) = \gamma(R)^{-1}$
2. $\gamma(R \circ S) = \gamma(R) \circ \gamma(S)$

*A homomorphism $\gamma : \Gamma \to \Gamma'$ is called a monomorphism if $\gamma(R) = \gamma(S)$ implies $R = S$ for all relations $R, S$ and an isomorphism if $\gamma(R) = \gamma(S) \iff R = S$.*

**Definition 3.** *For two relation algebras $\Gamma, \Gamma'$ a tractability-preserving-homomorphism (tph) is a homomorphism $\gamma$ from $\Gamma$ to $\Gamma'$ such that each subset $\beta \subseteq \Gamma$ is tractable iff $\gamma(\beta) \subseteq \Gamma'$ is tractable. An isomorphic tph is called tpi.*

Of course, tph and tpi are reflexive and transitive. In the literature there can be implicitly found a tph on $PA_{po}$ [4] and another one for directed intervals [13].

**Definition 4.** *A coarsening of a relation algebra is a monomorphism on its set of relations which is not an isomorphism.*

An example of a coarsening of RCC-8 is RCC-5 (cf. Fig. 3). The tph and tpi are important for problems like the following: Given are two relation algebras $\Gamma$ and $\Gamma'$. If we have a tph $\gamma : \Gamma \xrightarrow{\gamma} \Gamma'$ and know that $\beta \subseteq \Gamma$ is (in)tractable, then we know that

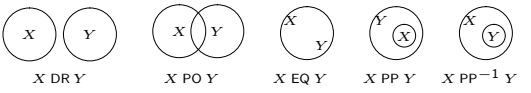


**Fig. 3.** The RCC-5 Relations

the image $\gamma(\beta)$ is (in)tractable in $\Gamma'$. But, how do we know if a subset $\beta'$ of $\Gamma'$ is the image of a tractable subset $\beta$ of $\Gamma$ ? The advantage of a tpi compared to a tph is that for a tpi there exists an inverse function, so that it is possible to transfer the tractable classes in both directions. If we know a maximal tractable subclass $\beta' \subseteq \Gamma'$, the tph does not allow to say anything about the maximal tractable subclasses of $\Gamma$. But with a tpi, we could identify the maximal tractable subclasses of $\Gamma$.

In order to identify maximal tractable subclasses of $PA_{dc}$, the tph given in [4] is here not sufficient. Instead, we use the following tpi $\gamma$:

**Lemma 2.** *A tpi $\gamma$ from $PA_{dc}$ to RCC-5 (cf. Fig. 3) is given by:*

$$\begin{array}{lll} \gamma : \prec \mapsto \mathsf{PP} & = \mapsto \mathsf{EQ} & \succ \mapsto \mathsf{PP}^{-1} \\ \curlywedge \mapsto \mathsf{PO} & \asymp \mapsto \mathsf{DR} & \end{array}$$

*Proof.* The claim that $\gamma$ is a homomorphism follows from a comparision of the composition table of $PA_{dc}$ with RCC-5 and the observation that for the only non-symmetric relation $\prec$ holds: $\gamma(\prec)^{-1} = (\mathsf{PP})^{-1} = \mathsf{PP}^{-1} = \gamma(\succ)$. Therefore, it is sufficient to prove that $\gamma$ is tractability-preserving and isomorphic. The reduction is an one-to-one identification of the basic relations in linear time. As any relation of a relation algebra consists of a union of basic relations, we can identify each relation of $PA_{dc}$ with a relation of RCC-5 in linear time. Therefore, the reduction is isomorphic and tractability-preserving if it preserves satisfiability.

In order to show that for each model of $PA_{dc}$ there is a model in RCC-5, let a set of constraints of $PA_{dc}$ with model $M$ be given. For each element $y$ in $M$, we introduce a point $P_y$. We define for each element $x$ in $M$ a set $S_x = \{P_y \mid y \prec x\}$ so that:

$$\begin{array}{lll} S_x \subsetneq S_y & & \iff x \prec y \\ S_x \supsetneq S_y & & \iff x \succ y \\ S_x = S_y & & \iff x = y \\ S_x \cap S_y = \emptyset & \iff \neg\, \exists a :\, a \in S_x \cup S_y & \iff x \asymp y \\ S_x \cup S_y \notin \{S_x, S_y\} & & \\ \quad \wedge\, S_x \cap S_y \neq \emptyset & \iff \quad \exists a :\, a \in S_x \cap S_y & \iff x \curlywedge y. \end{array}$$

So, the reduction from $PA_{dc}$ to RCC-5 preserves the satisfiability. Finally, it has to be shown that a model of RCC-5 is a model of $PA_{dc}$. We interprete the $PA_{dc}$ relations as derived from the partial order $\subseteq = \{\mathsf{PP}, \mathsf{EQ}\}$.

$$\begin{array}{lllll} X\ \mathsf{PP}\ Y & \iff & X\ \{\mathsf{PP}, \mathsf{EQ}\}\ Y \quad \wedge \neg(X\ \mathsf{EQ}\ Y) & \implies & X \prec Y \\ X\ \mathsf{EQ}\ Y & \iff & & & X = Y \\ X\ \mathsf{PP}^{-1}\ Y & \iff & X\ \{\mathsf{PP}, \mathsf{EQ}\}^{-1}\ Y \wedge \neg(X\ \mathsf{EQ}\ Y) & \implies & X \succ Y \end{array}$$

$X\ \mathsf{PO}\ Y$ implies $\exists Z \;\; Z \subseteq X \wedge Z \subseteq Y$. From this, it follows that there is a $Z$ with $Z \prec X$ and $Z \prec Y$, and this is the definition of $\curlywedge$. Similarly, from $X\ \mathsf{DR}\ Y$ follows $\neg\, \exists Z \;\; Z \subseteq X \wedge Z \subseteq Y$, hence $X \asymp Y$.

The tpi is not only an isomorphism between RCC-5 and $PA_{dc}$, but also a polynomial reduction in both directions. For that reason, the following theorem is an easy implication:

**Theorem 1.** *SAT($PA_{dc}$) is NP-complete.*

With Lemma 1, we get the next consequence.

**Corollary 1.** *NDP($PA_{dc}$) is NP-complete.*

The tpi provides us not only with the results about the NP-completeness of the general satisfiability problem, but also allows to identify all maximal tractable subclasses by a reduction to the classes presented in [12]. The results are depicted in Table 2.

**Table 2.** The tractable subclasses of $PA_{dc}$

| | $\tau_{28}$ | $\tau_{20}$ | $\tau_{17}$ | $\tau_{14}$ | | $\tau_{28}$ | $\tau_{20}$ | $\tau_{17}$ | $\tau_{14}$ |
|---|---|---|---|---|---|---|---|---|---|
| $\bot$ | • | • | • | • | $\{=\}$ | • | • | • | • |
| $\{\asymp\}$ | • | • | | | $\{=,\asymp\}$ | • | • | • | |
| $\{\curlywedge\}$ | • | • | | | $\{=,\curlywedge\}$ | • | • | • | |
| $\{\asymp,\curlywedge\}$ | • | • | | | $\{=,\curlywedge,\asymp\}$ | • | • | • | |
| $\{\prec\}$ | • | | | • | $\{=,\prec\}$ | • | | • | • |
| $\{\asymp,\prec\}$ | • | • | | | $\{=,\prec,\asymp\}$ | • | • | • | |
| $\{\curlywedge,\prec\}$ | • | | | | $\{=,\prec,\curlywedge\}$ | • | | • | |
| $\{\asymp,\curlywedge,\prec\}$ | • | • | | | $\{=,\prec,\curlywedge,\asymp\}$ | • | • | • | |
| $\{\succ\}$ | • | | | • | $\{=,\succ\}$ | • | | • | • |
| $\{\asymp,\succ\}$ | • | • | | | $\{=,\succ,\asymp\}$ | • | • | • | |
| $\{\curlywedge,\succ\}$ | • | | | | $\{=,\succ,\curlywedge\}$ | • | | • | |
| $\{\asymp,\curlywedge,\succ\}$ | • | • | | | $\{=,\succ,\curlywedge,\asymp\}$ | • | • | • | |
| $\{\prec,\succ\}$ | | | | • | $\{=,\prec,\succ\}$ | | | • | • |
| $\{\asymp,\prec,\succ\}$ | | • | | • | $\{=,\prec,\succ,\asymp\}$ | | • | • | • |
| $\{\curlywedge,\prec,\succ\}$ | • | | | • | $\{=,\prec,\succ,\curlywedge\}$ | • | | • | • |
| $\{\asymp,\curlywedge,\prec,\succ\}$ | • | • | | • | $\top$ | • | • | • | • |

In fact, the relations including $\{=\}$ are contained in $\tau_{28}$, $\tau_{20}$, $\tau_{14}$ if and only if $R \setminus \{=\}$ is contained in $\tau_{28}$ ($\tau_{20}$,$\tau_{14}$). $\tau_{17}$ contains all relations including $\{=\}$ and the empty relation $\bot$.

**Theorem 2.** *The four classes $\tau_{28}, \tau_{20}, \tau_{17}, \tau_{14}$ (cf. Tab. 2) are the only maximal tractable subclasses of $PA_{dc}$.*

Such correspondence functions have clear advantages: Not only do we have the answers for all questions concerning the complexity investigations, and we know that RCC-5 and $PA_{dc}$ are satisfiability equivalent up to isomorphism, but we have also found a connection between the field of reasoning on directed graphs and on topological constraint problems. Since there are many techniques known for handling graphs, some benefits for the RCC-5 algebra by such a translation should be possible.

# 4 The Associated Interval Algebra

## 4.1 Definition of the Interval Algebra $IA_{dc}$

There are applications in which it is not sufficient to compare single points in a network. For instance, pollution in a pipe network is not restricted to single points but extends to whole sections, and automated planning and project management deal with tasks that span over time intervals. This shows the need for a calculus with intervals as its basic elements. The basic relations are the relations between intervals that are definable by $PA_{dc}$ relations of its endpoints (cf. Fig. 4).

**Definition 5.** *An* interval $I = [s_I, e_I]$ *is a pair of points satisfying* $s_I \prec e_I$. *The interval algebra* $IA_{dc}$ *is the relation algebra generated by quadruples of relations as basic relations*

$$\mathcal{B} = \left\{ \begin{pmatrix} R_{ss} & R_{se} \\ R_{es} & R_{ee} \end{pmatrix} \mid R_{ss}, R_{se}, R_{es}, R_{ee} \in \{\prec, =, \succ, \asymp, \curlywedge\} \right\}$$

*closed under* $\cap, \cup, ^{-}, \circ, ^{-1}$. *For* $I = [s, e]$ *and* $I' = [s', e']$, *being in relation* $I\ R\ I'$ *means* $s\, R_{ss}\, s',\ s\, R_{se}\, e',\ e\, R_{es}\, s'$, *and* $e\, R_{ee}\, e'$.

In an identical way, the associated interval algebra $IA_{po}$ for partial order $PA_{po}$ can be introduced based on the set $\{\prec, =, \succ, \|\}$. The concepts and results for $IA_{dc}$ are also applicable to $IA_{po}$ as for $IA_{po}$, the two relations $\{\curlywedge, \asymp\}$ collapse into one relation $\{\|\}$. Some examples for $IA_{dc}$ relations are given in Fig. 5. In $IA_{po}$, the three relations in the upper row collapse to the relation $fa = \left(\begin{smallmatrix} \| & \| \\ \succ & \succ \end{smallmatrix}\right)$ and in the lower row to $fa^{-1}$.

In the following, we write $(R)$ for $\left(\begin{smallmatrix} R & R \\ R & R \end{smallmatrix}\right)$ $(R \in PA_{dc})$ and $eq$ for $\left(\begin{smallmatrix} = & \prec \\ \succ & = \end{smallmatrix}\right)$. Further, $\left(\begin{smallmatrix} \{R_1, R_2\} & R_{se} \\ R_{es} & R_{ee} \end{smallmatrix}\right)$ abbreviates $\left(\begin{smallmatrix} R_1 & R_{se} \\ R_{es} & R_{ee} \end{smallmatrix}\right) \cup \left(\begin{smallmatrix} R_2 & R_{se} \\ R_{es} & R_{ee} \end{smallmatrix}\right)$, etc.

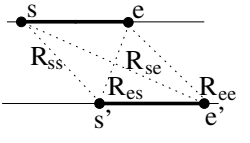


**Fig. 4.** A basic $IA_{dc}$ relation is a combination of the basic $PA_{dc}$ relations $R_{ss}, R_{se}, R_{es}, R_{ee}$

Fig. 5 and Fig. 6 show advantages of the new, finer interval calculus: More situations can be distinguished and more conclusions are possible. Fig. 6 shows a situation in a railway network. An obstruction B blocks the way from $A$ to $C$. Can $A$ reach its aim $C$? The situation is formally described by: $A\ (\prec)\ A_1,\ A_1\ (\prec)\ C,\ A_2\ (\prec)\ C,\ B\ (\prec)\ C,$

$$B\left(\begin{smallmatrix} \{\prec, =, \succ\} & \prec \\ \{\prec, =, \succ\} & \prec \end{smallmatrix}\right) A_1, \quad B\left(\begin{smallmatrix} \{\asymp, \curlywedge\} & \prec \\ \{\asymp, \curlywedge\} & \prec \end{smallmatrix}\right) A_2, \quad A_1\left(\begin{smallmatrix} \{\asymp, \curlywedge\} & \prec \\ \succ & = \end{smallmatrix}\right) A_2$$

By specifying the latter relation, e.g. to $A_1\left(\begin{smallmatrix} \asymp & \prec \\ \prec & = \end{smallmatrix}\right) A_2$, new conclusions can be drawn, in this case, $A\ (\prec)\ A_2$ becomes impossible. This means, if $B$ is an obstacle on the path from $A$ to $C$ via $A_1$, then there is no alternative route via $A_2$.

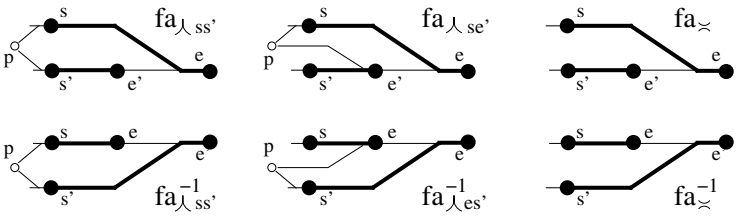


**Fig. 5.** Six examples for $IA_{dc}$ relations. Refinements of the "finally after" relation and its inverse.

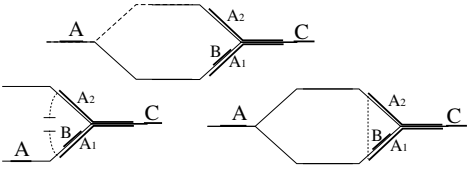


**Fig. 6.** Reasoning with uncertainty. The dashed line indicates that it is unknown if there is a path from $A$ to $A_2$. In contrast to $IA_{po}$, $IA_{dc}$ concludes such knowledge: Depending on the $IA_{dc}$ relation between $A_1$ and $A_2$, the interval $B$ can or cannot be bypassed using direction $A_2$.

*Remark 1.* $IA_{dc}$ is a finite relation algebra consisting of $45$ basic relations.

For each $\|$ occurring as a point relation in an $IA_{po}$ relation, specifiying it to either $\asymp$ or $人$, leads to different $IA_{dc}$ relations. Omitting impossible cases like $\left(\begin{smallmatrix} 人 & \asymp \\ * & * \end{smallmatrix}\right)$, $\left(\begin{smallmatrix} 人 & * \\ \asymp & * \end{smallmatrix}\right)$, $\left(\begin{smallmatrix} 人 & * \\ * & \asymp \end{smallmatrix}\right)$, $\left(\begin{smallmatrix} * & 人 \\ * & \asymp \end{smallmatrix}\right)$, $\left(\begin{smallmatrix} * & * \\ 人 & \asymp \end{smallmatrix}\right)$ leaves $45$ possible cases, where $*$ can be any basic relation.

For $IA_{dc}$, the set $\mathcal{U}$ of unions of basic relations with $\cup, \cap$, and $^-$ forms a boolean algebra. For basic relations, the operators $\circ$ and $^{-1}$ lead to the following relations of $\mathcal{U}$:

$$\begin{pmatrix} R_{ss} & R_{se} \\ R_{es} & R_{ee} \end{pmatrix}^{-1} = \begin{pmatrix} {R_{ss}}^{-1} & {R_{es}}^{-1} \\ {R_{se}}^{-1} & {R_{ee}}^{-1} \end{pmatrix} \tag{3}$$

$$R \circ R' = \begin{pmatrix} R_{ss} \circ R'_{ss} \wedge R_{se} \circ R'_{es} & R_{ss} \circ R'_{se} \wedge R_{se} \circ R'_{ee} \\ R_{es} \circ R'_{ss} \wedge R_{ee} \circ R'_{es} & R_{es} \circ R'_{se} \wedge R_{ee} \circ R'_{ee} \end{pmatrix} \tag{4}$$

Because of equations (1) and (2), $\mathcal{U}$ contains the result of operations $\circ, ^{-1}$ on arbitrary relations. Fig. 7 gives examples for such compositions.

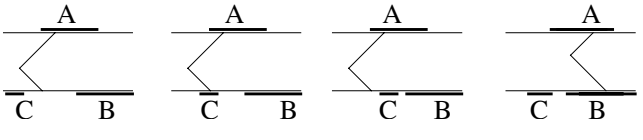


**Fig. 7.** In contrast to $IA_{po}$, $IA_{dc}$ discerns these specifications of $A(\{\asymp, 人\})B \circ B(\succ)C$

### 4.2 The Complexity of the Interval Algebra $IA_{dc}$

For a given set of $IA_{dc}$ constraints, how hard is it to decide the satisfiability?

**Theorem 3.** *The satisfiability problem of $IA_{dc}$ is NP-hard.*

*Proof.* (Sketch) We give a tpi from $PA_{dc}$ to a subset of $IA_{dc}$. We set $\gamma(=) = eq$ and $\gamma(R) = \left(\begin{smallmatrix} R & R \\ R & R \end{smallmatrix}\right)$ for all other basic relations, and extend it by $\gamma(R \cup R') = \gamma(R) \cup \gamma(R')$. By applying the rules (3) and (4) above, it is possible to show that $\circ, ^{-1}$ are preserved. For any solution of the interval case, a solution of the point case is found by picking an arbitrary point from the interval. Vice versa, for every solution of the point algebra, a solution of the interval case can be constructed by "inflating" the points, i.e. for each point $x$, we introduce an interval $\bar{x} = [x^-, x^+]$. As satisfiability is preserved and the translation can be done in linear time, the reduction is tractability-preserving.

Thus, for specific subclasses of $IA_{dc}$ we know if they are tractable, namely for the ones containing only unions of relations of the type (R) or $eq$. But, are there larger tractable subclasses? We define special subclasses of unions of basic relations: *pointizable* relations $\mathcal{P}$ and *gadgetable* relations $\mathcal{G}$. The class of pointizable relations has been for the first time introduced for $IA_{lin}$, and shown to be a tractable subclass [10]. However, we will show that the general class is not tractable for $IA_{dc}$ and neither for $IA_{po}$.

**Definition 6.** *For a subset $\mathcal{S} \subseteq PA_{dc}$, a relation $R$ is called $\mathcal{S}$-pointizable if it belongs to the class*

$$\mathcal{P}_S = \left\{ \left(\begin{smallmatrix} R_{ss} & R_{se} \\ R_{es} & R_{ee} \end{smallmatrix}\right) | R_{es}, R_{se}, R_{es}, R_{ee} \in \mathcal{S} \right\}$$

$\mathcal{P}_\mathcal{S}$ consists of those $IA_{dc}$ relations that can exactly be expressed by a set $\mathcal{S}$ of $PA_{dc}$ relations between its endpoints. What if we generalize this concepts to $IA_{dc}$ relations that can exactly be expressed by a set of $PA_{dc}$ relations between a larger set of points? A superset of the endpoints with such relations is called a gadget.

**Definition 7.** *A* gadget $(V_m, E_m)$ *for an $IA_{dc}$ relation $R$ is a set of point variables $V_m = \{p_1, \ldots, p_m\}$ $(m \geq 4)$ with $PA_{dc}$ relations between them so that:*

1. *In each satisfying assignment of $(V_m, E_m)$, $[p_1, p_2]R[p_3, p_4]$ is satisfied.*
2. *For each assignment of $p_1, \ldots, p_4$ holds: It can be extended to $p_1, \ldots, p_m$ in a way satisfying all relations of $(V_m, E_m)$ iff $[p_1, p_2]R[p_3, p_4]$ is satisfied.*

*For $\mathcal{S} \subseteq PA_{dc}$, an $IA_{dc}$ relation $R$ is called* $\mathcal{S}$-gadgetable *($R \in \mathcal{G}_\mathcal{S}$) if all the $PA_{dc}$ relations of a gadget for $R$ are relations of $\mathcal{S}$. We write $\mathcal{G}$ for $\mathcal{G}_{PA_{dc}}$ and $\mathcal{P}$ for $\mathcal{P}_{PA_{dc}}$.*

*Example.* With $I = [p_1, p_2]$ and $I' = [p_3, p_4]$, the points $p_1, .., p_6$ with the relations indicated in Figure 8 form a gadget for the $IA_{dc}$ relation $I \left(\begin{smallmatrix} \asymp & \prec \\ \succ & \{\prec, \curlywedge\} \end{smallmatrix}\right) I'$.

Consider the first property: $p_1 \asymp p_3$ is directly a relation of the gadget. The point $p_5$ with its relations enforces that $p_1 \prec p_4$, $p_2 \succ p_3$, and $p_2 \not\asymp p_4$. As $p_6$ with its relations enforces that $p_2 \neq p_4, p_2 \not\succ p_4$, only $p_2\{\prec, \curlywedge\}p_4$ remains satisfiable. The second property holds since any model of $I \left(\begin{smallmatrix} \asymp & \prec \\ \succ & \{\prec, \curlywedge\} \end{smallmatrix}\right) I'$ can be extended to include $p_5, p_6$ in a way satisfying all these $PA_{dc}$ relations.

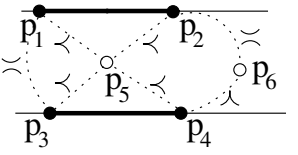


**Fig. 8.** A gadget with 6 points (with $PA_{dc}$ relations as indicated at the dotted edges)

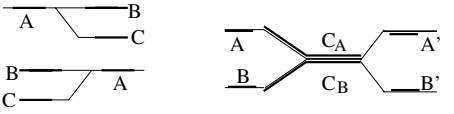


**Fig. 9.** The left part shows the two only cases satisfying the gadget for $A\,(\prec) \cup (\succ)\,B$, and the right part shows the two possibilities $(A, B)$ and $(A', B')$ satisfying the gadget for $(\curlywedge) \cup (\asymp)$

All basic $IA_{dc}$ relations and relations $(R)$ are pointizable, and pointizability is closed under $\cap$ and $^{-1}$, but not under $\cup$. For instance, the relation $(\prec) \cup (\succ)$ [1] is not pointizable. But all relations are subsets of a pointizable relation because

$$\begin{pmatrix} R_{ss} & R_{se} \\ R_{es} & R_{ee} \end{pmatrix} \cup \begin{pmatrix} R'_{ss} & R'_{se} \\ R'_{es} & R'_{ee} \end{pmatrix} \subseteq \begin{pmatrix} R_{ss} \cup R'_{ss} & R_{se} \cup R'_{se} \\ R_{es} \cup R'_{es} & R_{ee} \cup R'_{ee} \end{pmatrix}.$$

**Theorem 4.** *$\mathcal{G}, \mathcal{P}$ are intractable subclasses of $IA_{dc}$.*

*Proof.* By definition, $\mathcal{P}_\mathcal{S} \subseteq \mathcal{G}_\mathcal{S}$. Consider the set $\mathcal{R} := \{(\{\asymp, \curlywedge\}), (\{\prec, \succ\}), \left(\begin{smallmatrix} \succeq & \preceq \\ \succ & = \end{smallmatrix}\right)\}$. We show that $\mathcal{R} \subsetneq \mathcal{P}$ is intractable. Due to Table 2, $\{\{\asymp, \curlywedge\}, \{\prec, \succ\}\}$ is not contained in a tractable subclass of $PA_{dc}$. Applying $\gamma$ from Theorem 3 shows that the satisfiability problem over $\mathcal{I} := \{(\asymp) \cup (\curlywedge),\ (\prec) \cup (\succ)\} \subsetneq IA_{dc}$ is intractable. A satisfiability problem over $\mathcal{I}$ can be reduced to $\mathcal{R}$ because $A(\prec) \cup (\succ)B$ is satisfiable iff the corresponding gadget $A(\{\prec, \succ\})B$, $A(\{\prec, \succ\})C$, $B(\{\asymp, \curlywedge\})C$ is satisfiable, and $A(\asymp) \cup (\curlywedge)B$ is satisfiable iff $A(\{\asymp, \curlywedge\})B$, $A(\prec) \cup (\succ)C_A$, $B(\prec) \cup (\succ)C_B$, $C_A\left(\begin{smallmatrix} \succeq & \preceq \\ \succ & = \end{smallmatrix}\right)C_B$ is satisfiable (cf. Fig. 9). Hence, intractability is inherited from $\mathcal{I}$ via $\mathcal{R}$ to $\mathcal{P}$ and $\mathcal{G}$.

Not all gadgetable relations are pointizable. For instance, $(\prec) \cup (\succ)$ is gadgetable, but not pointizable: Hence $\mathcal{B} \subsetneq \mathcal{P} \subsetneq \mathcal{G} \subsetneq IA_{dc}$.

If $\mathcal{S}$ is tractable in $PA_{dc}$, $\mathcal{P}_\mathcal{S}$ is tractable in $IA_{dc}$. How about the larger class $\mathcal{G}_\mathcal{S}$?

**Theorem 5.** *If $\mathcal{R}$ is a class of $\mathcal{S}$-gadgetable $IA_{dc}$ relations and the satisfiability problem over $\mathcal{S}$ is tractable, then the satisfiability problem over $\mathcal{R}$ is tractable.*

*Proof.* For a given set of constraints for $n$ intervals, we need to find a polynomial algorithm that decides if it is satisfiable. The idea is to translate the $\mathcal{S}$-gadgetable $IA_{dc}$ constraints into corresponding $PA_{dc}$ constraints over $\mathcal{S}$. Since $\mathcal{S}$ is a tractable subclass of $PA_{dc}$, there is a polynomial procedure to decide consistency. We have to show that only a polynomial number of $PA_{dc}$ relations is needed, and that the translation is possible in polynomial time. For each of the finitely many interval relations, a gadget is fixed. The

[1] This relation may describe that two projects need to be done by the same person in any order.

largest one defines an upper bound $M$ for the used gadget's size (number of points in it). Hence, replacing the $\mathcal{S}$-gadgetable $IA_{dc}$ constraints by the associated $\mathcal{S}$ constraints can be done in $O((M \cdot n^2)^2) = O(n^4)$.

**Corollary 2.** *$\mathcal{B}, \mathcal{G}_{\tau_{28}}, \mathcal{G}_{\tau_{20}}, \mathcal{G}_{\tau_{17}}, \mathcal{G}_{\tau_{14}}$ are tractable subclasses of $IA_{dc}$.*

$\mathcal{B}$ is tractable since $\mathcal{B} \subseteq \mathcal{G}_{\{\{\prec\},\{=\},\{\succ\},\{\asymp\},\{\wedge\}\}} \subseteq \mathcal{G}_{\tau_{28}}$.

From this, the NP-completeness of $IA_{dc}$ can be concluded. The NP-hardness follows directly from Theorem 3. The membership in NP follows from the tractability of constraints of basic relations. For an arbitrary set of constraints of $IA_{dc}$, we guess a scenario (i.e. all constraints are basic relations), and because of Corollary 2, we can test if it is satisfiable. With Lemma 1, we get the complexity class of the $IA_{dc}$ problems.

**Corollary 3.** *SAT($IA_{dc}$) and NDP($IA_{dc}$) are NP-complete.*

## 5 Outlook and Summary

Starting from the question how a relation algebra useful for reasoning about dependencies in general networks should be designed, we have identified a calculus that qualifies points in such a network. This calculus consists of five basic relations.

Traditional reasoning for points in networks uses only four relations: before, equal, after, and unrelated. This language has been used so far for modelling networks usually thought of as models for relational systems in space, time or space-time. We showed that an algebra for reasoning about dependencies needs a new 'fork' relation. This relation, which states that two points have a common ancestor, proved useful in various application areas from ecological systems, over transportation networks, to planning and medical diagnosis. Also dependencies in a Bayesian network can be expressed.

In a formal analysis of this calculus, we proved that the complexity of the general satisfiability problem is NP-complete as well as the satisfiability problem of the corresponding interval algebra $IA_{dc}$. This means, in general, the problem of finding out if there is a network satisfying certain conditions, is rather difficult. But in many cases, there is a polynomial algorithm. We have identified all tractable subclasses of $PA_{dc}$, via a tractability preserving isomorphism into the RCC-5 calculus. Classes of the new type of $\mathcal{S}$-gadgetable relations were identified as tractable subclasses of $IA_{dc}$. We have shown how these results for $IA_{dc}$ can also be applied to the proper subclass $IA_{po}$.

This work opens the field in many directions: By the tph and tpi techniques presented here, the satisfiability equivalence of different relation algebras can be shown, and the expressibility and complexity of these algebras can be transfered and compared. Further investigations should reveal that the class of $\mathcal{S}$-gadgetables are maximal tractable subclasses for $IA_{po}$ and $IA_{dc}$. Variations of the presented calculi are possible. Making the similar distinction for common consequences like for common causes could easily be expressed by our calculus with a direction-reversed interpretation. The combination of both provides a framework to satisfy the suggested extension of the linear directed intervals on networks [13]. Another promising idea is the temporalization of the dependency calculus for modeling dependencies that vary over time.

## Acknowledgments

This work was partially supported by the Deutsche Forschungsgemeinschaft (DFG) as part of the Transregional Collaborative Research Center SFB/TR 8 Spatial Cognition. We like to thank Bernhard Nebel for various helpful discussions.

## References

1. J. F. Allen. Maintaining knowledge about temporal intervals. *Comm. ACM*, 26(11): 832–843, 1983.
2. F. Anger, P. Ladkin, and R. Rodriguez. Atomic temporal interval relations in branching time: Calculation and application. In *Actes 9th SPIE Conference on Applications of AI*, Orlando, FL, USA, 1991.
3. M. Broxvall and P. Jonsson. Towards a complete classification of tractability in point algebras for nonlinear time. In *Proc. of CP-99*: 129–143, 1999.
4. M. Broxvall, P. Jonsson, and J. Renz. Refinements and Independence: A Simple Method for Identifying Tractable Disjunctive Constraints. In *CP*: 114–127, 2000.
5. A.G. Cohn. Qualitative spatial representation and reasoning techniques. In KI-97: Advances in AI, Brewka, G. and Habel, C. and Nebel, B (eds), LNAI, 1–30, 1997.
6. T. Drakengren and P. Jonsson. A complete classification of tractability in Allen's algebra relative to subsets of basic relations. *Artificial Intelligence*, 106(2): 205–219, 1998.
7. C. Freksa. Using Orientation Information for Qualitative Spatial Reasoning. In Theories and Methods of Spatial-Temporal in Geographic Space. Reasoning. Frank, A. U. and Campari, I. and Formentini, U. (eds.), 162–178, 1992.
8. P. B. Ladkin and R. D. Maddux. On binary constraint problems. *J. ACM*, 1994.
9. U. Montanari. Networks of constraints: Fundamental properties and applications to picture processing. *Inform. Sci.*, 7:95-132,1974.
10. B. Nebel and H.-J. Bürckert. Reasoning about temporal relations: A maximal tractable subclass of Allen's interval algebra.*J.ACM*, 42(1):43–66, 1995.
11. Randell, D. and Cui, Z. and Cohn, A. A Spatial Logic Based on Regions and Connection. Proceedings KR-92, 165–176, 1992.
12. J. Renz and B. Nebel. On the complexity of qualitative spatial reasoning: A maximal tractable fragment of the Region Connection Calculus. *AIJ*, 108(1-2):69–123, 1999.
13. J. Renz. A Spatial Odyssey of the Interval Algebra: 1. Directed Intervals. In *Proc. of IJCAI'01*, 2001.
14. M. B. Vilain, H. A. Kautz, and P. G. van Beek. Contraint propagation algorithms for temporal reasoning: A revised report. *Reasoning about Physical Systems*: 373–381, 1989.