Point Algebras for Temporal Reasoning: Algorithms and Complexity

Mathias Broxvall and Peter Jonsson
Department of Computer and Information Science
Linköping University
S-581 83 Linköping, Sweden
e-mail: {Mathias.Broxvall,Peter.Jonsson}@ida.liu.se

Abstract

We investigate the computational complexity of temporal reasoning in different time models such as totally-ordered, partially-ordered and branching time. Our main result concerns the satisfiability problem for point algebras and point algebras extended with disjunctions—for these problems, we identify all tractable subclasses. We also provide a number of additional results; for instance, we present a new time model suitable for reasoning about systems with a bounded number of unsynchronized clocks, we investigate connections with spatial reasoning and we present improved algorithms for certain problems.

1 Introduction

Reasoning about temporal knowledge is a common task in computer science and elsewhere, cf. Golumbic and Shamir [23] for an extensive list of examples from a wide variety of application areas. In most applications, knowledge of temporal constraints is expressed in terms of collections of relations between time points and/or time intervals. Typical reasoning tasks include determining the satisfiability of such collections and deducing new relations from those that are known.

Research on automated reasoning about temporal constraints has largely concentrated on simple, linear models of time and the computational properties of such constraints have been investigated in depth. For example, it is known that the point algebra is tractable [39] and all tractable fragments of the point-interval and interval-interval relations have been identified [26, 28]. However, it has been observed many times that more complex time models are needed in, for instance, the analysis of concurrent and distributed systems, certain planning domains, robot motion problems and cooperating agents [2, 12, 15, 19, 20, 29, 31, 40]. Typical problems include the traditional difficulties of synchronization and communication problems, network and bus protocol correctness and decision making in the absence of a global clock. Some concrete examples of these problems are presented in [38].

Consequently, a number of alternative time models have been proposed and the two most well-known are *partially-ordered* time and *branching* time. The partially-ordered model of time has mainly been used for studying distributed systems (e.g. cooperating agents) [2, 30]. The branching time model has been used, for instance, in planning [15, 31] and in the analysis of concurrent systems [20]. It should also be noted that several logics based on branching time (such as CTL and CTL*) have been thoroughly investigated, cf. the tutorial by Emerson and Srinivasan [21]. Furthermore, the point algebra for branching time has been examined by Düntsch *et al.* [18] from an algebraic point of view. Other examples of interesting time models include the *parallel worlds* model [29], *relativistic time* [30] and the *directed intervals* model [36].

Most of the previous research on non-standard time models have concentrated on modeling properties and theoretical foundations while computational aspects have received relatively little attention. There are a few papers that address such questions, though. Clarke *et al.* [12] have provided an apparently efficient algorithm than can check the correctness of certain circuits and algorithms. The method builds on a propositional branching-time logic based on CTL (Computation Tree Logic). CTL (or extended versions of it) is frequently used in model-checking—a good survey can be found in [32]. Anger *et al.* [4] have studied whether standard constraint-based approaches can be applied to reasoning about nonlinear time; their results show that this is possible in certain restricted cases. Computational properties of the point algebra for branching time has been investigated by Hirsch [24]. He shows, among other things, that the full point algebra is tractable (satisfiability can be decided by an $O(n^5)$ time algorithm) but k-consistency (for arbitrary k) is insufficient for determining satisfiability. This can be compared to many other relational algebras where path consistency decides consistency for the tractable sets of relations, for instance, the point algebra for totally ordered time [39] and all known tractable subclasses of RCC-5 [11] and RCC-8 [35].

The aim of this paper is to study different time models from a computational perspective. More specifically, we do the following:

1. We identify all tractable subclasses of the point algebra for partially-ordered time and we identify all tractable subclasses of the point algebra extended with disjunctions for partially-ordered, totally-ordered and branching time. Note that it has been proven earlier that the full point algebras for totally-ordered and branching time are tractable [39, 24].

2. We argue that a certain time model, *partially ordered time with bounded dimension*, is suitable when reasoning about systems containing a finite number of unsynchronized clocks. We show that reasoning in this model is unusually hard when the dimension is > 1 (since even the fragment only containing basic relations is NP-complete) and give a full classification of tractability for the point algebra as well as for the point algebra extended with disjunctions.

3. We demonstrate how our results can be used for proving complexity results in other domains. As a concrete example, we use our results on partially-ordered time for studying a simple spatial reasoning formalism (whose objects consists of non-degenerate two-dimensional convex sets). By exhibiting a connection between these two formalisms, we arrive at a complete classification of tractability in the spatial formalism almost for free.

4. Finally, we give an improved algorithm for the full point algebra for branching time running in $O(nM(n))$ time where $M(n)$ is the time complexity of multiplying two $n \times n$ integer matrices. Coppersmith and Winograd [14] have shown that there exists such an algorithm running in $O(n^{2.376})$ time. By using their algorithm, our algorithm runs in $O(n^{3.376})$ time which is comparable to that of path consistency checking algorithms, $O(n^3)$. It is thus a significant improvement over Hirsch's [24] $O(n^5)$ algorithm.

The reader may rightfully ask why we give so much attention to disjunctive extensions of point algebras. Instead of considering all possible subclasses of a temporal algebra, one can concentrate on subclasses that are 'interesting' in some sense. Whether disjunctive subclasses are interesting or not is of course a matter of taste but we can provide some evidence that they are worth studying. First, simple constraint languages extended with disjunctions have historically proved to have interesting properties. For instance, the Horn DLRs [25] which subsumes almost all previously presented temporal languages for total-ordered time can be viewed as a point algebra extended with disjunctions. Several other similar examples are given in Cohen *et al.* [13]. Secondly, disjunctions can compactly describe complex relations. Consider for example the ORD-Horn algebra [33] which is a tractable subclass of Allen's algebra. It contains 868 different relations and is consequently quite difficult to remember. Defining the ORD-Horn with the aid of disjunctions is much easier: ORD-Horn contains exactly the Allen relations which can be expressed by disjunctions of the form $x_1 \leq y_1 \vee x_2 \neq y_2 \vee \ldots \vee x_n \neq y_n$. Interval algebras for time models other than totally-ordered time have been studied in [3]. They show, for instance, that there are 19 and 29 basic relations in the interval algebra for branching and partially-ordered time, respectively. The large number of relations (2^{19} and 2^{29}) in the full algebras clearly motivate concise descriptions of tractable subclasses.

Another reason is that we would like to answer open questions posed by Broxvall *et al.* [10]. Their paper provides necessary and sufficient conditions for tractability of two types of disjunctive constraint languages ($\Gamma \overset{\times}{\vee} \Delta^*$ and $\Gamma \cup \Delta^2$, the exact definitions can be found in Section 2). One of the open questions is whether there exists tractable disjunctive languages of other 'types' (such as $\Gamma \overset{\times}{\vee} \Delta$ is tractable, but $\Gamma \overset{\times}{\vee} \Delta^*$ is not) or not. To answer such questions, we need to thoroughly analyze disjunctive constraints. In an attempt to be systematic in our search, we consider point algebras with very different computational properties.

1. tractable point algebras that are solvable by enforcing path-consistency (totally-ordered time);
2. tractable point algebras that are not solvable by enforcing path-consistency (branching time);
3. intractable point algebras with a tractable fragment containing all basic relations (partially-ordered time); and
4. intractable point algebras without a tractable fragment containing all basic relations (partially-ordered time with bounded dimension).

The findings made in this article supports the conjecture that there are just a few classes of maximal tractable constraint languages - no 'structurally' new constraint classes were discovered.

Yet another reason is that disjunctive constraint classes can effectively be embedded in different logics (where the disjunction operator typically is a key ingredient in the logic language). Very clear examples of this approach has been given by Drakengren and Bjäreland [16, 17] who show how expressive temporal logics for reasoning about action and change can be based on disjunctive constraints.

The paper is structured as follows: Section 2 contains basic definitions and some auxiliary results. Sections 3–6 contains the results for totally- and partially-ordered time, partially ordered time with bounded dimension and branching time, respectively. Finally, some concluding remarks are collected in Section 7. Parts of this article have previously appeared in three conference papers [7, 8, 9].

2 Preliminaries

2.1 Point algebras

A point algebra is based on the notion of *relations* between pairs of variables interpreted over a partially-ordered set. We consider four *basic relations* which we denote by $<, >, =$ and $\|$. If x, y are points in a partial order $(T, \leq)$, then we define these relations in terms of the partial ordering $\leq$ as follows:

1. $x < y$ iff $x \leq y$ and not $y \leq x$
2. $x > y$ iff $y \leq x$ and not $x \leq y$
3. $x = y$ iff $x \leq y$ and $y \leq x$
4. $x \| y$ iff neither $x \leq y$ nor $y \leq x$

These basic relations are a set of jointly exhaustive and pairwise disjoint (JEPD) relations. The complete set of relations we consider are disjunctions of basic relations and they are represented as unions of basic relations. Since we have

4 different basic relations we get $2^4 = 16$ possible relations. The set of basic relations is denoted $\mathcal{B}$ and the set of all 16 relations is denoted by $\mathcal{PA}$. We use a self-explanatory, short-hand notation for relations, for example, $< \cup =$ is written as $\leq$ and $= \cup \parallel$ as $\underline{\parallel}$. The empty relation is denoted by $\perp$ and is usually omitted in our definitions of the tractable sets of relations for clarity. Note that a tractable set of relations extended with the empty relation is still tractable.

The basic computational problem is the satisfiability problem where we have a set of variables, a set of constraints over the variables and the question is whether there exists a mapping from the variables to some partial order such that all constraints are satisfied.

Definition 1 Let $X \subseteq \mathcal{PA}$ be a set of point relations and p a class of partial orders. A problem instance of $\text{PSAT}_p(X)$ is a set of variables V and a set of binary constraints of the form xry where $x, y \in V$ and $r \in X$. A tuple $\langle f, \langle T, \leq \rangle \rangle$ where $f : V \to T$ is a total function and $(T, \leq) \in P$ is called an interpretation of Π.

A problem instance Π is satisfiable iff there exists an interpretation $M = (f, (T, \leq))$ such that $f(x)\ R\ f(y)$ holds for every constraint xRy. Such an M is called a *model* of Π.

We note once and for all that $\text{PSAT}_P(X)$ is in NP for all choices of P and X considered in this paper. Given a problem instance $\Pi = (V, C)$, let $\text{Var}(\Pi) = V$. By slightly abusing the notation, we write $c \in \Pi$ to denote the fact that the constraint c is a member of C. The size of a problem instance can either be regarded as the total number of variables and constraints or (as is common when studying algorithms for enforcing path consistency) simply the total number of variables. We choose the later approach since this gives stronger time bounds than the alternative definition.

A point algebra $\mathcal{A}$ consists of the 16 possible relations between points together with the operations *converse* $\cdot^{-1}$, *intersection* $\cap$ *and composition* $\circ$ which are defined as follows:

$$\forall x, y : \ xr^{-1}y \Leftrightarrow yrx$$

$$\forall x, y : \ x(r \cap s)y \Leftrightarrow xry \text{ and } xsy$$

$$\forall x, y : \ x(r \circ s)y \Leftrightarrow \exists z : \ (xrz \text{ and } zsy)$$

It follows that the converse of $r = b_1 \cup \ldots \cup b_n$ is equal to $b_1^{-1} \cup \ldots \cup b_n^{-1}$. The intersection of two relations can be expressed as the usual set-theoretic intersection. Using the definition of composition, it can be derived that given two relations $r = b_1 \cup \ldots \cup b_n$ and $r' = b'_1 \cup \ldots \cup b'_m$,

$$r \circ r' = \bigcup \{ b_i \circ b'_j \mid i \leq n \text{ and } j \leq m \}.$$

Given a set X of relations, we define the *closure* of X, $C(X)$, as the least set X' of relations closed under converse, intersection and composition and having the property $X \subseteq X'$. We say that $X \subseteq \mathcal{PA}$ is a *subalgebra* if $C(X) = X$. We have the following result [37].

Theorem 2 $\textsc{Psat}_P(X)$ is tractable (NP-complete) if and only if $\textsc{Psat}_P(C(X))$ is tractable (NP-complete).

We say that a set of relations X is a *maximal tractable* subclass iff $\textsc{Psat}_p(X)$ is tractable and there exist no X' such that (1) $X \subset X' \subseteq \mathcal{PA}$; and (2) $\textsc{Psat}_p(X')$ is tractable. If X is a maximal tractable subclass, then $C(X) = X$.

2.2 Disjunctions

We will now show how to extend point algebras with disjunctions. This section contains the basic definitions together with some results needed for proving the classification theorems.

Definition 3 Let R_1, R_2 be two relations of arity i, j over a domain D and define the disjunction $R_1 \vee R_2$ of arity $i + j$ as follows:

$$R_1 \vee R_2 = \{(x_1, \dots, x_{i+j}) \in D^{i+j} | (x_1, \dots, x_i) \in R_1 \vee (x_{i+1}, \dots, x_{i+j}) \in R_2\}$$

Thus, the disjunction of two relations with arity i, j is the relation with arity $i + j$ satisfying either of the two relations. To give a concrete example of how the disjunction operator is used, consider the following example.

Example 2.1 Let $D = \{0, 1\}$ and the relations $\mathsf{And} = \{(1, 1)\}$ and $\mathsf{Xor} = \{(0, 1), (1, 0)\}$ be given. The disjunction of And and Xor is:

$$\mathsf{And} \vee \mathsf{Xor} = \left\{ \begin{array}{c} (0,0,0,1),(0,1,0,1),(1,0,0,1),(1,1,0,1), \\ (0,0,1,0),(0,1,1,0),(1,0,1,0),(1,1,1,0), \\ (1,1,0,0),(1,1,0,1),(1,1,1,0),(1,1,1,1) \end{array} \right\}$$

We see that the constraint x And $y \vee x$ Xor z encoded by $(\mathsf{And} \vee \mathsf{Xor}, x, y, x, z)$ is satisfied when x, y and z have been instantiated to, for instance $1, 0, 0$ respectively.

Let Γ_1, Γ_2 be sets of relations and define the disjunction of two sets of relations $\Gamma_1 \overset{\times}{\vee} \Gamma_2$ as follows:

$$\Gamma_1 \overset{\times}{\vee} \Gamma_2 = \Gamma_1 \cup \Gamma_2 \cup \{R_1 \vee R_2 | R_1 \in \Gamma_1, R_2 \in \Gamma_2\}$$

The disjunction of two sets of relations $\Gamma_1 \overset{\times}{\vee} \Gamma_2$, first introduced by Cohen *et al.* [13], is the set of disjunctions of each pair of relations in Γ_1, Γ_2 plus the sets Γ_1, Γ_2. It is natural to include Γ_1 and Γ_2 since one wants to have the choice of using the disjunction or not. The fact that if $R_1 \vee R_2$ is included in a set of relations, then both R_1 and R_2 are in the set is an property which we refer to as the $\overset{\times}{\vee}$-closure property. In the sequel, we will tacitly assume that all sets of relations have this property.

We shall frequently be concerned with constraints that are specified by disjunctions of an arbitrary number of relations. Thus, we make the following definition: for any set of relations, Δ, define $\Delta^* = \bigcup_{i=0}^{\infty} \Delta^i$ where $\Delta^0 = \{\bot\}$ and $\Delta^{i+1} = \Delta^i \overset{\times}{\vee} \Delta$.

The previously defined concepts of problem instances, models and so on can obviously be extended to disjunctions in a natural way. It is also worth noticing that the problem $\textsc{Psat}_p$ is still in NP if we allow the use of disjunctions.

We say that a set of relation Γ is *maximal tractable* for $\textsc{Psat}_p$ iff $\textsc{Psat}_p(\Gamma)$ is tractable and for every set $X \not\subseteq \Gamma$ of relations from $\mathcal{PA}^*$ it holds that $\textsc{Psat}_p(X)$ is not tractable. Given a complete classification of $\textsc{Psat}_p$ with disjunctions, we can easily find a complete classification of the point algebra over p.

Theorem 4 Let M be the set of maximal tractable sets of relations for $\textsc{Psat}_p(\mathcal{PA}^*)$. Then, the set $M' = \{X \cap \mathcal{PA} \mid X \in M\}$ contains every maximal tractable set of $\textsc{Psat}_p(\mathcal{PA})$.

Proof: Let $X' \subseteq \mathcal{PA}$ be a maximal tractable set. There exists an $X \in M$ such that $X' \subseteq X$ since M contains the maximal tractable sets of $\mathcal{PA}^*$ (and $\mathcal{PA} \subseteq \mathcal{PA}^*$). Hence, $X' \subseteq X \cap \mathcal{PA}$. It cannot be the case that $X' \subset X \cap \mathcal{PA}$ since $X \cap \mathcal{PA}$ is tractable and X' is a maximal tractable set. Consequently, $X' = X \cap \mathcal{PA}$ and $X' \in M'$. $\square$

It follows immediately that the maximal tractable sets of $\textsc{Psat}_p(\mathcal{PA})$ are the maximal elements of the partial order $(M', \subseteq)$.

Next, we introduce a construction which simplifies maximality proofs by allowing us to only consider a small number of disjunctions. Let Γ and Δ be arbitrary sets of relations such that $\Delta \subseteq \Gamma$.

Definition 5 Let $\mathcal{T} = \Gamma \overset{\times}{\vee} \Delta^*$. We define $\overline{\mathcal{T}}$ as

$$(\mathcal{PA} - \Gamma) \cup (\Gamma - \Delta) \overset{\times}{\vee} (\Gamma - \Delta) - (\Gamma - \Delta).$$

Lemma 6 If $\mathcal{T} = \Gamma \overset{\times}{\vee} \Delta^*$ and $\mathcal{T}' \not\subseteq \mathcal{T}$, then there exists $C \in \overline{\mathcal{T}}$ such that $C \in \mathcal{T}'$.

Proof: Arbitrarily choose a $C \in \mathcal{T}'$ such that $C \notin \mathcal{T}$ and choose $C_1, \cdots, C_n \in \mathcal{PA}$ such that $C_1 \vee \cdots \vee C_n = C$. Assume first that there exists some i such that $C_i \notin \Gamma$. The definition of $\overline{\mathcal{T}}$ implies that $C_i \in \overline{\mathcal{T}}$ and the $\overset{\times}{\vee}$-closure property implies that $C_i \in \mathcal{T}'$.

Assume instead that $C_1, \cdots, C_n \in \Gamma$. Since $\Gamma \subseteq \mathcal{T}$ and $C \notin \mathcal{T}$ we know that $n > 1$. If all or all but one $C_i \in \Delta$, then we know that $C \in \mathcal{T}$. Hence, there exists at least two i, j such that $C_i, C_j \notin \Delta$. Now, $C_i \vee C_j \in \mathcal{T}'$ by the $\overset{\times}{\vee}$-closure property and the definition of $\overline{\mathcal{T}}$ gives that $C_i \vee C_j \in \overline{\mathcal{T}}$. $\square$

For the tractability results we will use the concept of *k-independence* [13]. This property is defined as follows

Definition 7 For any sets of relations Γ and Δ, define $\textsc{Psat}_{p,\Delta \leq k}(\Gamma \cup \Delta)$ to be the subproblem of $\textsc{Psat}_p(\Gamma \cup \Delta)$ consisting of all instances containing at most k constraints over the relations in Δ. We say that Δ is *k-independent* of Γ if the following condition holds: any set of constraints C in $\textsc{Psat}_p(\Gamma \cup \Delta)$ has a solution provided every subset of C belonging to $\textsc{Psat}_{p,\Delta \leq k}(\Gamma \cup \Delta)$ has a solution.

We will refer to 1-independence as simply independence. The following result by Cohen *et al.* [13] demonstrates the usefulness of the k-independence property:

Theorem 8 Let Γ and Δ be sets of relations such that $\mathrm{PSAT}_p(\Gamma \cup \Delta)$ is tractable. If Δ is 1-independent of Γ, then $\mathrm{PSAT}_p(\Gamma \overset{\times}{\vee} \Delta^*)$ is tractable. If Δ is 2-independent of $\emptyset$, then $\mathrm{PSAT}_p(\Delta \overset{\times}{\vee} \Delta)$ is tractable.

Example 2.2 We see that $\{=\}$ is not 1-independent of $\{=, \neq\}$ since the constraints $X = \{x = y, y = z, x \neq z\}$ is not satisfiable although the constraints $X' = \{x = y, x \neq z\}$ and $X'' = \{y = z, x \neq z\}$ are both satisfiable. Note that $\{=\}$ is 1-independent of $\{=\}$ since all problem instances containing only equality constraints are satisfiable.

We continue by proving that the 1-independence property is preserved under the closure operator.

Theorem 9 Let Δ be a set of relations 1-independent of Γ. Then, $C(\Delta)$ is 1-independent of $C(\Gamma)$.

Proof: We begin by showing that Δ is 1-independent of $C(\Gamma)$. By inspecting Renz and Nebel's [37] proof of Theorem 2, it follows that any instance of $\mathrm{PSAT}_p(C(\Gamma))$ can be transformed to an equivalent instance of $\mathrm{PSAT}_p(\Gamma)$. In fact, each constraint xRy where $R \in C(\Gamma) - \Gamma$ is merely replaced by a fixed instance of $\mathrm{PSAT}_p(\Gamma)$ over the variables x, y and (possibly) a number of auxiliary variables. It follows that Δ is 1-independent of $C(\Gamma)$ since Δ is 1-independent of Γ.

Now, let Π be an arbitrary instance of $\mathrm{PSAT}_p(C(\Gamma) \cup C(\Delta))$ and let S be the subinstance of Π only containing relations from $C(\Gamma) - C(\Delta)$. Assume that every subset $S \cup \{x_1 \delta_1 y_1\}, \ldots, S \cup \{x_n \delta_n y_n\}$ of Π where $\delta_i \in C(\Delta)$ is satisfiable. Each such set of constraints can be rewritten on the equivalent form $S \cup \Pi_i$ where Π_i is a CSP instance over Δ. Obviously, each set $S \cup \Pi_i$ is satisfiable as well as all the relaxed sets $S \cup \{\pi_{i,1}\}, \ldots, S \cup \{\pi_{i,m}\}$, where $\{\pi_{i,1}, \ldots, \pi_{i,m}\}$ are the constraints in Π_i. Since Δ is 1-independent of $C(\Gamma)$ and Π_i only contains relations from Δ, we know that $S \cup H$ where $H = \{\pi_{i,j} | i \leq n, j \leq m\}$ is satisfiable. Since $S \cup H$ is equivalent to the instance Π, we have shown that Π is satisfiable and, hence, that $C(\Delta)$ is 1-independent of $C(\Gamma)$. $\square$

For the total classifications in Section 3–6 we need to prove that the satisfiability problem $\mathrm{PSAT}_p(\{R_1, R_2, R_3 \vee R_3\})$ is NP-hard for different instantiations of p, R_1, R_2 and R_3. For this purpose, we provide a result inspired by Broxvall *et al.* [10]. Note that $\mathrm{PSAT}_p(\{R_1, R_2\} \cup \{R_3\}^2)$ can be reduced to $\mathrm{PSAT}_p(\{R_1, R_2, R_3 \vee R_3\})$ in polynomial time.

Theorem 10 $\mathrm{PSAT}_p(\Gamma \cup \Delta^2)$ is tractable only if Δ is 2-independent of Γ. Otherwise, $\mathrm{PSAT}_p(\Gamma \cup \Delta^2)$ is NP-complete.

Proof: Assume to the contrary that $\text{PSAT}_p(\Gamma \cup \Delta^2)$ is tractable but Δ is not 2-independent of Γ. This implies that there exists a set of constraints X over Γ and a set $H = \{h_1, \ldots, h_n\}$ of constraints over Δ such that $X \cup \{h_i, h_j\}$ is satisfiable for every $1 \leq i, j \leq n$ but $X \cup H$ is not satisfiable. Choose X and H such that $|H|$ is as small as possible and note that $|H| \geq 3$. The existence of a set $H' \subset H$ such that $X \cup H'$ is not satisfiable contradicts the minimality of H so $X \cup H'$ is satisfiable for all $H' \subset H$. Thus, we can define the satisfiable set $\mathcal{X} = X \cup \{h_1 \vee h_2, h_1 \vee h_3, h_2 \vee h_3, h_4, \ldots, h_n\}$ which have the following property: In every model of $\mathcal{X}$, exactly one of h_1, h_2, h_3 is not satisfied.

To prove the result, we show that 3-COLOURABILITY can be transformed to $\text{PSAT}_p(\Gamma \cup \{R \vee R\})$ in polynomial time. Arbitrarily choose an undirected graph $G = (V, E)$ such that $V = \{v_1, \ldots, v_k\}$. We will construct an instance of $\text{PSAT}_p(\Gamma \cup \{R \vee R\})$ that is satisfiable iff G is colourable with three colours.

For each vertex v_i, introduce a fresh copy of the set $\mathcal{X}$ where we denote the constraints h_1, h_2, h_3 as h_1^i, h_2^i, h_3^i, respectively. As we have already noted, this will force exactly one of h_1^i, h_2^i, h_3^i not to hold in every model. We interpret that h_j^i does not hold as 'vertex v_i has colour j'.

For each edge $(v_i, v_j) \in E$, we add the disjunctions $h_1^i \vee h_1^j$, $h_2^i \vee h_2^j$ and $h_3^i \vee h_3^j$ which ensures that v_i and v_j are not assigned the same colour. The resulting set of constraints can be computed in polynomial time, it is an instance of $\text{PSAT}_p(\Gamma \cup \Delta^2)$ and it is satisfiable iff G is 3-colourable which concludes the proof. $\square$

To simplify the presentation of some tractable sets of relations, we recall the definition of the *guaranteed satisfaction property* [13].

Definition 11 Let Δ be a set of relations. If every instance of $\text{PSAT}_p(\Delta)$ which does not contain a constraint of the form $x \bot y$ is satisfiable, then we say that Δ has the *guaranteed satisfaction* (GS) property.

Clearly, $\text{PSAT}_p(\Delta)$ and $\text{PSAT}_p(\Delta^*)$ are tractable if Δ has the GS property. Define the sets $\Delta_=$ and $\Delta_{\|}$ such that $\Delta_= = \{\bot\} \cup \{R \in \mathcal{PA} \mid (=) \subseteq R\}$ and $\Delta_{\|} = \{\bot\} \cup \{R \in \mathcal{PA} \mid (\|) \subseteq R\}$ Since we encounter these sets of relations frequently, we note once and for all that $\Delta_=$ and $\Delta_{\|}$ has the GS property. We also note that whenever an instance of $\text{PSAT}_p(\Delta_=)$ has a model, it has a one-point model. Similarly, a satisfiable instance of $\text{PSAT}_p(\Delta_{\|})$ always has a model where the points are unrelated. Also note that we can always extend a tractable set of relations with the equality relation, i.e. $\text{PSAT}_p(\Delta \cup \{=\})$ is tractable whenever $\text{PSAT}_p(\Delta)$ is tractable.

3 Totally Ordered Time

We will now identify the maximal tractable subclasses of the totally ordered point algebra. The domain we are considering is the set of total orders, *to*, and the corresponding decidability problem is denoted PSAT_{to}. The methods and the results presented in this section will be used many times in later sections when we classify point algebras over other types of partial orders.

Note that the basic relation $||$ is unnecessary when dealing with total orders so we only have three basic relations ($<$, $=$ and $>$) and eight possible disjunctions of these relations. Let $\mathcal{PA}_{to}$ denote the set of these eight relations and define $\mathcal{X}_1 = \mathcal{PA}_{to} \overset{\times}{\vee} \{\neq\}^*$ and $\mathcal{X}_2 = \Delta_{=}^*$. As we will see later on, $\mathcal{X}_1$ and $\mathcal{X}_2$ are the only maximal tractable subclasses of PSAT_{to}.

Lemma 12 $\text{PSAT}_{to}(\mathcal{X}_i)$, $1 \leq i \leq 2$, are tractable problems.

Proof: Tractability of $\mathcal{X}_1$ has been proved by Jonsson and Bäckström [25] and Koubarakis [27] while the tractability of $\mathcal{X}_2$ follows trivially since $\Delta_{=}$ has the GS property. □

Lemma 13 $\text{PSAT}_{to}(\mathcal{N}_i)$, $1 \leq i \leq 5$, is NP-complete where

$$\begin{array}{ll} \mathcal{N}_1 = \{< \vee <\} & \mathcal{N}_2 = \{\neq,\ \leq \vee \leq\} \\ \mathcal{N}_3 = \{<,\ = \vee =\} & \mathcal{N}_4 = \{\neq,\ = \vee =\} \\ \mathcal{N}_5 = \{<,\ \leq \vee \leq\}. & \end{array}$$

Proof: We begin by proving that Δ_i is not 2-independent of Γ_i for Δ_i, Γ_i as defined below:

$$\Delta_1 = \{<\}, \Delta_2 = \{\leq\}, \Delta_3 = \{=\}, \Delta_4 = \{=\}, \Delta_5 = \{\leq\}$$

$$\Gamma_1 = \emptyset, \Gamma_2 = \{\neq\}, \Gamma_3 = \{<\}, \Gamma_4 = \{\neq\}, \Gamma_5 = \{<\}$$

Together with Theorem 10, this proves that the correspodning satisfiability problems are NP-complete. The proof is straightforward: the following sets of constraints are not satisfiable although every subinstance of them containing at most two relations from Δ_i are satisfiable.

1. $x_1 < x_2 < x_3 < x_1$
2. $x_1 \leq x_2 \leq x_3 \leq x_1, x_1 \neq x_2$
3. $x_1 = x_2 = x_3 = x_4, x_1 < x_4$
4. $x_1 = x_2 = x_3 = x_4, x_1 \neq x_4$
5. $x_1 \leq x_2 \leq x_3 \leq x_1, x_1 < x_2$

□

We are now ready to prove the main theorem.

Theorem 14 $\mathcal{X}_1$ and $\mathcal{X}_2$ are the only maximal tractable disjunctive subclasses of PSAT_{to}.

Proof: Assume that there exists a maximal tractable algebra $\mathcal{X}$ such that $\mathcal{X} \not\subseteq \mathcal{X}_1$ and $\mathcal{X} \not\subseteq \mathcal{X}_2$. By Lemma 6, there exists γ_1, γ_2 in $\mathcal{T}$ such that $\gamma_1 \in \overline{\mathcal{X}_1}$ and $\gamma_2 \in \overline{\mathcal{X}_2}$. It is easy to see that $\overline{\mathcal{X}_1} \subseteq \{r_1, \ldots, r_6\}$ where

$$\begin{array}{rclrcl} r_1 & = & (< \vee <) & r_2 & = & (\leq \vee \leq) \\ r_3 & = & (= \vee =) & r_4 & = & (\leq \vee =) \\ r_5 & = & (< \vee =) & r_6 & = & (< \vee \leq) \end{array}$$

and $\overline{\mathcal{X}_2} \subseteq \{<, \neq, < \vee <, < \vee \neq, \neq \vee \neq\}$. By Lemma 22, we know that $\mathrm{PSAT}_{to}\{(< \vee <)\}$ is NP-complete so the disjunction $r_1 = (< \vee <)$ can be excluded immediately. We will show that if γ_1 equals one of $r_2, \ldots, r_6$ and γ_2 equals $<$ or $\neq$, then $\mathrm{PSAT}_{to}(\{\gamma_1, \gamma_2\})$ is NP-complete.

1. $\gamma_1 \in \{r_2, r_3\}$. Then, the NP-completeness of $\mathrm{PSAT}_{to}(\{\gamma_1, \gamma_2\})$ is an immediate consequence of Lemma 13.

2. $\gamma_1 = r_4$. Note that r_3 can be obtained from r_4 so $\mathrm{PSAT}_{to}(\{\gamma_1, \gamma_2\})$ is NP-complete by case 1.

3. $\gamma_1 = r_5$ and $\gamma_2 = <$. We show how to obtain the disjunctive relation $a < b \vee c < d$ with γ_1 and γ_2: introduce a fresh variable t and the following two relations: $a < t$ and $t = b \vee c < d$. NP-completeness follows from case 1.

4. $\gamma_1 = r_5$ and $\gamma_2 = \neq$. The relation $a < b$ can easily be obtained by the relations $t_1 \neq t_2$ and $t_1 = t_2 \vee a < b$ where t_1 and t_2 are fresh variables. NP-completeness follows from the previous case.

5. $\gamma_1 = r_6$. The NP-completeness of $\mathrm{PSAT}_{to}(\{\gamma_1, \gamma_2\})$ is a consequence of cases 3 and 4 and the observation that r_5 can be obtained from r_6.

Now, $<$ can be obtained from $< \vee \neq$ since $x < y \vee z \neq z$ (where z is a fresh variable) and $x < y$ are equivalent. Similarly, $\neq$ can be obtained from $\neq \vee \neq$. NP-hardness follows from the cases above so $\mathrm{PSAT}_{to}(\{\gamma_1, \gamma_2\})$ is NP-complete for all $\gamma_1 \in \overline{\mathcal{X}_1}$ and $\gamma_2 \in \overline{\mathcal{X}_2}$. This contradicts our initial assumptions and proves the theorem. $\Box$

4 Partially Ordered Time

We will now consider the class *po* of all partial orders. The results can be summarized as follows: the point algebra extended with disjunctions contains the following four maximal tractable subclasses:

$$\mathcal{S}_A = \Gamma_A \overset{\times}{\vee} \Delta_A^*, \; \mathcal{S}_B = \Gamma_B \overset{\times}{\vee} \Delta_B^*, \; \mathcal{S}_C = \Gamma_C \overset{\times}{\vee} \Delta_C^*, \; \mathcal{S}_D = \Delta_D^*.$$

The different sets of relations are defined in Table 2 and a composition table can be found in Table 1. An immediate consequence of Theorem 4 is that the point algebra (without disjunctions) contains three maximal tractable subclasses Γ_A, Γ_B and Δ_D.

	$<$	$>$	$=$	$\parallel$
$<$	$<$	$\mathcal{B}$	$<$	$<\parallel$
$>$	$\mathcal{B}$	$>$	$>$	$>\parallel$
$=$	$<$	$>$	$=$	$\parallel$
$\parallel$	$<\parallel$	$>\parallel$	$\parallel$	$\mathcal{B}$

Table 1: Composition of basic relations in partially-ordered time.

	Γ'_A	Δ'_A	Γ_A	Δ_A	Γ'_B	Δ'_B	Γ_B	Δ_B	Γ'_C	Δ'_C	Γ_C	Δ_C	Δ_D
$<$			•				•		•		•		
$\leq$	•		•		•		•						•
$<>$							•	•					
$<=>$					•	•	•	•					•
$\parallel$	•	•	•	•							•	•	
$\underline{\parallel}$	•		•								•	•	•
$=$			•				•		•		•		•
$\neq$	•	•	•	•	•	•	•	•	•	•	•	•	
$<\parallel$			•	•							•	•	
$\leq\parallel$			•						•	•	•	•	•

Table 2: Tractable classes in partially-ordered time.

We begin by making a number of observations. For the non-disjunctive case, Γ_A is the only maximal tractable set containing all basic relations. For the point algebra extended with disjunctions, $\mathcal{S}_A$ and $\mathcal{S}_C$ are the only maximal tractable sets containing all the basic relations. Furthermore, a satisfiable problem instance over Γ_B or $\mathcal{S}_B$ always has a model that is a total order; similarly, a satisfiable problem instance over Δ_D or $\mathcal{S}_D$ has a one-point model since $\mathcal{S}_D = \Delta_=$.

4.1 Tractability Results

We will now show that the four classes of disjunctive relations $\mathcal{S}_A, \ldots, \mathcal{S}_D$ (as defined in the beginning of Section 4) are tractable. We begin by showing that Γ_A, Γ_B and Δ_D are tractable. The tractability result follows from combining these results with a series of independence results and applying Theorem 8.

To prove the tractability of Γ_A and Γ_B, we present algorithm A (Figure 1) and claim that it correctly solves the problems $\textsc{Psat}_{po}(\Gamma'_A)$ (where $\Gamma'_A = \{\leq, \underline{\|}, \neq\}$) and $\textsc{Psat}_{po}(\Gamma'_B)$ (where $\Gamma'_B = \{\leq, <=>, \neq\}$). By proving that $C(\Gamma'_A) = \Gamma_A$ and $C(\Gamma'_B) = \Gamma_B$ (and using Theorem 2), we are done.

We need some definitions before we present the algorithm. Given an instance Π of $\textsc{Psat}_p(X)$ and two variables x, y we write $x \leq^+ y$ to say that there exists zero or more variables $z_1, \ldots, z_n$ such that

$$x \leq z_1 \wedge z_1 \leq z_2 \wedge \cdots \wedge z_{n-1} \leq z_n \wedge z_n \leq y$$

and we write $x \leq^* y$ to say $x \leq^+ y$ or $x = y$.

The *components* of a graph $G = (V, E)$ are the equivalence classes of vertices that are mutually reachable in the symmetric closure of G, i.e. the graph $G = (V, \{(v, w), (w, v) \mid (v, w) \in E\})$.

An R-subinstance Π' of Π is defined as $(V, C \cap (V \times \{R\} \times V))$. A variable v is *R-minimal* in the instance Π iff v is minimal in the R-subinstance of Π. The *R-components* of an instance Π are the components of the R-subinstance of Π. An instance Π is $(\leq)$-acyclic iff the $(\leq)$-subinstance of Π is acyclic.

If f is a function which is undefined for n, then $(f|n \longmapsto c)$ is defined as the function $f \cup \{(n, c)\}$. Assume n_1, n_2 are variables in an instance $\Pi = (V, C)$ and let $\Pi' = (V - \{n_2\}, C')$ where

$$C' = C \cup \{n_1 r x \mid n_2 r x \in C\} \cup \{x r n_1 \mid x r n_2 \in C\} - \{x r n_2, n_2 r x \mid x \in V\}.$$

We say that Π' is obtained by *contracting* n_1 and n_2. That is, we identify the variables n_1, n_2 by n_1. Note that there may appear constraints of the form xrx after a contraction. We will use a function of the form $\mathrm{contract}(\Pi, n_1, n_2)$ for computing the instance resulting from contracting the variables n_1, n_2 in Π.

The following simple result is needed several times for proving algorithm correctness.

Lemma 15 Let Π be an instance of $\mathrm{PSAT}_p(X)$ such that for every model M of Π it holds that $M(n_1) = M(n_2)$, and let Π' be the instance resulting from contracting the variables n_1, n_2 in Π. Then, Π is satisfiable iff Π' is satisfiable.

Proof: Assume that Π has a model M. Clearly M is also a model of Π'. Assume that Π' has a model $M' = (f, (T, \leq))$, the following is a model of Π:

$$M = ((f|n_2 \longmapsto f(n_1)), (T, \leq))$$

□

Theorem 16 $\mathrm{PSAT}_{po}(\Gamma_A)$ is tractable.

Proof: Algorithm A obviously runs in polynomial time since every loop removes at least one element from the given instance Π, so at most $|\Pi|$ loops can be made. We continue by showing that algorithm A solves $\mathrm{PSAT}_{po}(\Gamma'_A)$ correctly.

If the algorithm contracts two variables n_1, n_2 in an instance Π, then Π is satisfiable iff the contracted instance is satisfiable by Lemma 15 since either $n_1 \leq^* n_2, n_2 \leq^* n_1$ or $n_1 \| n_2, n_2 \leq^* n_1$ which both lead to $M(n_1) = M(n_2)$ in every model M of Π. Thus, if the algorithm rejects Π, then the original instance cannot have a model.

Assume to the contrary that the algorithm accepts an instance. Consider the instance Π after the last iteration and recall that the original instance is

algorithm $A(\Pi)$
Input: An instance Π of $\textsc{Psat}_{po}(\Gamma_A)$ or $\textsc{Psat}_{po}(\Gamma_B)$
 repeat
  $\Pi' \leftarrow \Pi$
  for each pair of variables $n_1, n_2 \in \mathrm{Var}(\Pi)$ **do**
   if $n_1 \| n_2$ and $n_1 \leq^* n_2$ in Π **then return false**
   if $n_1 \leq^* n_2$, $n_2 r n_1 \in \Pi$ and $r \in \{\leq^*, \underline{\|}\}$ **then**
    if $n_1 \neq n_2$ in Π **then return false**
    else $\Pi' \leftarrow$ contract(Π, n_1, n_2)
   end if
  end for
until $\Pi' = \Pi$
return true

Figure 1: The algorithm for solving $\textsc{Psat}_{po}(\Gamma'_A)$ and $\textsc{Psat}_{po}(\Gamma'_B)$

satisfiable iff Π is satisfiable. The $(\leq)$-subinstance of Π is acyclic and there exist no n_1, n_2 such that $n_1 \underline{\|} n_2$ and $n_1 \leq^* n_2$. Let,

$$M = (f, (T, \leq)) = (\{(n_i, p_i) | n_i \in \Pi\}, (\{p_i | n_i \in \Pi\}, \{(p_i, p_j) | n_i \leq^* n_j\}))$$

and note that $(T, \leq)$ is a partial order. Arbitrarily choose xRy in Π and assume without loss of generality that x, y are distinct variables. If R is $\neq$ the relation holds since every variable n_i is mapped to a unique point p_i, if R is $\underline{\|}$ the relation holds since neither $x \leq^* y$ nor $y \leq^* x$ in Π. The last case when R equals $\leq$ trivially holds. Consequently M is a model of Π and the original problem instance is satisfiable.

Finally, we note that $C(\Gamma'_A) = \Gamma_A$ since $(<) = (\leq \cap \neq)$, $(\leq \|) = (\leq \circ \underline{\|})$, $(< \|) = (\leq \| \cap \neq)$ and $(=) = (\leq \cap \geq)$. $\square$

Many different algorithms exist for solving $\textsc{Psat}_{po}(\Gamma_A)$. Anger *et al.* [4] have presented an alternative algorithm and Broxvall *et al.* [10] have shown that path consistency decides this problem. We have chosen to present algorithm A for mainly two reasons; (1) it shows how the structure of the partially-ordered time model can be exploited in an algorithm; and (2) the algorithm can be used for showing the independence results needed in the next section (where we study disjunctive constraints). The independence results can also be shown by an automatic method [10]; however, this method gives no clue whatsoever why the independence property holds. When we turn our attention to branching time in Section 6, the reader will benefit from having seen how an algorithm can be used for proving independence results—in that case, the automatic method does not work.

Theorem 17 $\textsc{Psat}_{po}(\Gamma_B)$ is tractable.

Proof: We show that algorithm A correctly solves the $\textsc{Psat}_{po}(\Gamma'_B)$ problem. If the algorithm contracts two variables n_1, n_2 in an instance Π, then Π is

algorithm B
Input: An instance Π of $\textsc{Psat}_{po}(\Gamma'_C)$
repeat
$\Pi' \leftarrow \Pi$.
if $\exists n_1, n_2 \in \mathrm{Var}(\Pi)$ s.t. $n_1 = n_2 \in \Pi$ **then** $\Pi \leftarrow \mathrm{contract}(\Pi, n_1, n_2)$
end if
until $\Pi' = \Pi$
if $\exists n \in \mathrm{Var}(\Pi)$ such that $n \neq n$ or $n < n$ in Π **then return false**
if $\exists n_1, n_2 \in \mathrm{Var}(\Pi)$ s.t. $n_1 <^* n_2 \in \Pi \wedge n_1 \; R \; n_2 \in \Pi$ where $< \notin R$ **then return false**
return true

Figure 2: The algorithm for solving $\textsc{Psat}_{po}(\Gamma'_C)$

satisfiable iff the contracted instance is satisfiable by Lemma 15 since there exists a cycle $n_1 \leq^* n_2 \leq^* n_1$ which leads to $M(n_1) = M(n_2)$ in every model M of Π. After having made this observation, it is trivial to show that if the algorithm rejects an instance Π, then it does not have a model.

Assume to the contrary that the algorithm accepts the instance Π and let Π' denote the instance after the last iteration of the algorithm. We can construct a model for Π' by induction over the number of variables in Π'. If Π' is empty then $M = (\emptyset, (\emptyset, \emptyset))$ is a model of Π'. Assume we can construct a model for every instance with k or fewer variables. Let Π' be an arbitrary instance containing $k+1$ variables. Note that the $(\leq)$-subinstance of Π' is acyclic and thus contains at least one minimal variable x. By the induction hypothesis there exists a model $M' = (f, (T, \leq))$ of $\Pi' - \{x\}$. Let,

$$M = ((f | n \longmapsto c), (T \cup \{c\}, \leq \cup \{(c, x) | x \in T\}))$$

where c is a fresh point. Clearly M is a model of Π' and thus Π is satisfiable.

To conclude the theorem, $C(\Gamma'_B) = \Gamma_B$ since $(<) = (\leq \cap \neq)$, $(=) = (\leq \cap \geq)$ and $(<>) = (<=> \cap \neq)$. $\square$

Observe that the model constructed in the previous proof is a total-order and whenever the algorithm rejects an instance, there is no partial order that can serve as a model for the instance. This proves the following corollary.

Corollary 18 $\textsc{Psat}_p(\Gamma_B)$ is tractable whenever $to \subseteq p$.

Finally, $\textsc{Psat}_{po}(\Delta_D)$ is a tractable problem since $\Delta_D = \Delta_=$. We can now continue with disjunctive constraints.

Lemma 19 $\Delta'_A, \Delta'_B, \Delta'_C$ are independent of Γ'_A, Γ'_B and Γ'_C, respectively.

Proof: We begin by showing that Δ'_A is independent of Γ'_A; the proof for Γ'_B is similar.

Let Π be an instance of $\mathrm{PSAT}_{po}(\Gamma'_A \cup \Delta'_A)$ and assume that every instance Π' such that Π' only contains constraints present in Π and at most one constraint from Δ'_A is satisfiable. We prove Π to be satisfiable.

Assume to the contrary that Π is not satisfiable. Then, there exists at least one constraint $x\ R\ y \in \Pi$ where R is $\neq$ or $\|$ which causes algorithm A to reject. Let Π' denote the problem instance containing all constraints from Π of the form $z\ R'\ w$ where R' is $\leq$ or $\underline{\|}$ plus the constraint $x\ R\ y$. Note that Π' causes the algorithm to reject. This is a contradiction, since Π' is a problem instance previously assumed to be satisfiable and the algorithm correctly solves $\mathrm{PSAT}_{po}(\Gamma'_A \cup \Delta'_A)$. Hence, Δ'_A is independent of Γ'_A.

To prove that Δ_C is independent of Γ'_C, we present an alternative tractable algorithm for Γ'_C. Obviously. $\Gamma_C \subseteq \Gamma_A$ so we could in principle use algorithm A—it is, however, easier to prove the result this way. A polynomial-time algorithm for $\mathrm{PSAT}_{po}(\Gamma'_C)$ is presented in Figure 2. Obviously algorithm B rejects only unsatisfiable instances of $\mathrm{PSAT}_{po}(\Gamma'_C)$.

Assume algorithm B accepts a certain instance Π and let Π' be the instance resulting from the contractions made by the algorithm in lines 3-6. Observe that Π' does not contain the relation $(=)$ and if Π' is satisfiable, then Π is satisfiable. Let I be the interpretation given by the function $f(x) = x$ and the partial order $(V, \{(x,y) \mid x <^* y \text{ in } \Pi'\})$ where V is the set of variables appearing in Π'.

Each $\neq$- and $<$-constraint is satisfied by I and if there exists a constraint $x \leq \| y$ in Π' then either $x < y$ or $x \| y$ under I which guarantees that each such constraint is satisfied so I is a model and Π is satisfiable. Thus, $\mathrm{PSAT}_{po}(\Gamma'_C)$ is tractable.

Proving that Δ'_C is independent of Γ'_C with the aid of algorithm B is analogous to the previous independence proofs. $\square$

We can now show that the four classes are tractable.

Theorem 20 $\mathrm{PSAT}_{po}(\mathcal{S}_A), \ldots, \mathrm{PSAT}_{po}(\mathcal{S}_D)$ are tractable problems.

Proof: That $\mathrm{PSAT}_{po}(\Gamma'_A \overset{\times}{\vee} \Delta'^*_A)$ is tractable follows from Lemmata 16, 19 and Theorem 8. That $\Gamma_A = C(\Gamma'_A)$ was shown in the proof of Theorem 16. Let Π be an arbitrary instance of $\mathrm{PSAT}_{po}(\mathcal{S}_A)$. We will show how the relation $< \|$ can be replaced by relations in $\Gamma'_A \overset{\times}{\vee} \Delta'^*_A$ which implies the tractability of $\mathrm{PSAT}_{po}(\mathcal{S}_A)$ by Theorem 9.

Choose an arbitrary disjunction $\gamma = x\ R_1\ y \vee \ldots \vee x_n\ R_n\ y_n$ in Π where $R_1 = (< \|)$. Introduce a fresh variable $t_{x,y}$, add the constraint $x \leq t_{x,y}$ and replace γ by the disjunction $\gamma' = t_{x,y} \parallel y \vee \ldots \vee x_n\ R_n\ y_n$. Repeat this transformation until no $(< \|)$ remains—a process which clearly takes polynomial time. Let Π' denote the resulting instance. Since the $(\leq \circ \|) = < \|$, the constraints $x \leq t_{x,y}, t_{x,y} \| y$ implies the relation $x < \| y$. It follows that Π' is satisfiable iff Π is satisfiable and that $\mathrm{PSAT}_{po}(\mathcal{S}_A)$ is tractable.

The tractability of $\mathrm{PSAT}_{po}(\Gamma'_B \overset{\times}{\vee} \Delta'^*_B)$ follows from Theorem 17, Lemma 19 and Theorem 8, and that $C(\Gamma'_B) = \Gamma_B$ was shown in the proof of Theorem 17. The two constraints $x \neq y$, $x <=> y$ are satisfiable iff $x <> y$ which shows that

$C(\Delta'_B) = \Delta_B$. This fact combined with Theorem 9 proves that $\textsc{Psat}_{po}(\mathcal{S}_B)$ is tractable.

To show that $\mathcal{S}_C$ is tractable we begin by noting that $C(\Delta'_C) = \Delta_C$ since $(\underline{\|}) = (\leq \| \ \cap \ (\leq \|)^{-1})$, $(\|) = (\neq \cap (\leq \|) \ \cap \ (\leq \|)^{-1})$ and $< \| = (\neq \cap \leq \|)$. Hence, Δ_C is independent of Γ'_C by Lemma 19 and Theorem 9. That $\textsc{Psat}_{po}(\Gamma'_C \overset{\times}{\vee} \Delta^*_C)$ is tractable follows from Theorem 8. The theorem follows since $\Gamma_C = \Gamma'_C \cup \Delta_C$ which implies that $\mathcal{S}_C = \Gamma'_C \overset{\times}{\vee} \Delta^*_C = \Gamma_C \overset{\times}{\vee} \Delta^*_C$.

Finally, $\textsc{Psat}_{po}(\mathcal{S}_D)$ is tractable since Δ_D has the GS property. $\square$

4.2 Intractability Results

This section contains the NP-completeness results needed for the classification theorem. Our hardness results rely on the fact that $\textsc{Psat}_{po}(\mathcal{A}_2)$ is NP-complete where $\mathcal{A}_2 = \{<>, \|\}$. We show hardness for NP by a polynomial-time reduction from the BETWEENNESS problem which is defined as follows:

INSTANCE: A pair (A, T) consisting of a finite set A and a collection T of ordered triples (a, b, c) of distinct elements from A.
QUESTION: Is there a one-to-one mapping $f : A \to \{1, 2, \ldots, |A|\}$ such that for each $(a, b, c) \in T$ we have either $f(a) < f(b) < f(c)$ or $f(c) < f(b) < f(a)$.

The BETWEENNESS problem is known to be NP-complete [22, page 279].

Given an arbitrary instance (A, T) of BETWEENNESS where $A = \{a_1, \ldots, a_k\}$, construct an instance Π of $\textsc{Psat}_{po}(\mathcal{A}_2)$ as follows:

1. for each $a_i \in A$, introduce a fresh variable a'_i;
2. for each pair of distinct $a_i, a_j \in A$, add the constraint $a_i <> a_j$;
3. for each triple $t_m = (a_i, a_j, a_k) \in T$, introduce two fresh variables x'_m, y'_m and add the following constraints:

$$x'_m \| a'_i,\ x'_m \| a'_j,\ x'_m <> a'_k$$

$$y'_m <> a'_i,\ y'_m \| a'_j,\ y'_m \| a'_k$$

We show that Π is satisfiable iff (A, T) has a solution. For the if-direction, assume that (A, T) is satisfiable and let $\mathcal{P}_{|A|}$ be the partial order $(V, <)$ where $V = \{x_1, \ldots, x_{|A|}, y_1, \ldots, y_{|A|}, z_1, \ldots, z_{|A|}\}$ and:

$$x_i < y_j, y_i < y_{j+1}, y_i < z_j \text{ iff } i \leq j$$
$$x_i \| x_j, x_i < z_j, z_i \| z_j \text{ for all } i, j$$

Furthermore, let f be a mapping from A to $\{1, 2, \ldots, |A|\}$ satisfying each triple in T. We build a mapping f' from the variables in Π as follows:

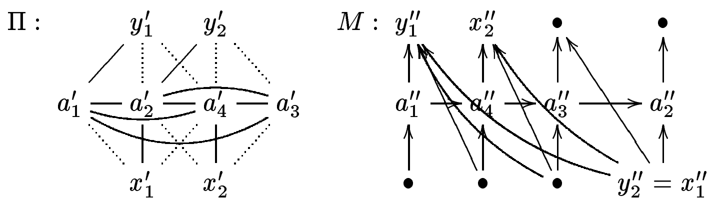

Figure 3: The problem instance Π and its model M. To simplify the picture, dotted arrows have been used to mark the relation $||$ and filled arrows for the relation $<>$

1. $f'(a'_i) = y_l$ where $f(a_i) = l$
2. For each $t_m = (a_i, a_j, a_k) \in T$ such that $f(a_i) < f(a_j) < f(a_k)$ we let $f'(x'_m) = x_{f(a_k)}, f'(y'_m) = z_{f(a_i)}$
3. For each $t_m = (a_i, a_j, a_k) \in T$ such that $f(a_i) > f(a_j) > f(a_k)$ we let $f'(x'_m) = z_{f(a_k)}, f'(y'_m) = x_{f(a_i)}$

If we examine all the constraints in Π it is obvious that f' satisfies them all.

To clarify the steps of the model construction above we provide a small example. The BETWEENNESS problem instance

$$(\{a_1, a_2, a_3, a_4\}, \{(a_1, a_4, a_2), (a_2, a_3, a_4)\})$$

gives the point algebra problem instance Π in figure 3 and has a solution $g = \{(a_1, 1), (a_2, 4), (a_3, 3), (a_4, 2)\}$. From the solution of the BETWEENNESS problem we can construct a model M of Π depicted in the same figure. Transitive edges are omitted.

We continue by showing the only-if direction. Assume that Π has a model M. Note that the variables $a'_1, a'_2, \ldots$ in Π corresponding to variables $a_1, a_2, \ldots$ in A are totally ordered, i.e. for arbitrary distinct a'_i, a'_j, either $M(a'_i) < M(a'_j)$ or $M(a'_j) < M(a'_i)$. Choose indices $m_1, \ldots, m_k$ such that $a'_{m_1} < \cdots < a'_{m_k}$. Define a solution $f : A \to \{1, \ldots, k\}$ of (A, T) as follows: $f(a_i) = m_i$. Clearly, this function is a one-to-one mapping from A to $\{1, \ldots, k\}$.

Arbitrarily choose a triple $(a_i, a_j, a_k) \in T$. We show that $f(a_i) < f(a_j) < f(a_k)$ or $f(a_k) < f(a_j) < f(a_i)$. Assume to the contrary that $f(a_j) < f(a_i) < f(a_k)$. Then, the model M implies that $M(a'_j) < M(a'_i) < M(a'_k)$ under the partial order of M. By the construction of Π we know that there exists a variable y'_m such that $M(y'_m) <> M(a'_i)$, $M(y'_m)||M(a'_j)$ and $M(y'_m)||M(a'_k)$. Assume first that $M(y'_m) < M(a'_i)$, then $M(y'_m) < M(a'_k)$ which contradicts the fact that $M(y'_m)||M(a'_k)$. Similarly, if $M(y'_m) > M(a'_i)$ then $M(a'_j) < M(y'_m)$ and we have a contradiction since $M(y'_m)||M(a'_j)$. The remaining three cases can be proven analogously.

Thus, we have proved the main part of the following lemma.

Lemma 21 If X is a set of relations containing $<>$ and one of $\|, < \|, \underline{\|}$ or $\leq \|$, then $\text{PSAT}_{po}(X)$ is NP-complete.

Proof: The case when $(\|) \in X$ was proved above. Assume $< \|$ or $\underline{\|}$ is in X. Then, $(\|) = (< \|) \cap (< \|)^{-1}$ or $(\|) = (<> \circ \underline{\|}) \cap \underline{\|}$. If X contains $\leq \|$, then $(\underline{\|}) = (\leq \|) \cap (\leq \|)^{-1}$ and NP-completeness follows from the previous case. $\square$

We also need a number of hardness results for disjunctive relations.

Lemma 22 The following problems are NP-complete:

1. $\text{PSAT}_{po}(\{< \vee <\})$;
2. $\text{PSAT}_{po}(\{R_1, \leq \vee \leq\})$ if $R_1 \in \{\neq, <>\}$;
3. $\text{PSAT}_{po}(\{R_2, = \vee =\})$ if $=$ is not a subset of R_2;
4. $\text{PSAT}_{po}(\{R_3, R_4, \underline{\|} \vee \underline{\|}\})$ if $R_3 \in \{\leq, <=>\}$ and $R_4 \in \{<, <>, \neq, \| <\}$.

Proof: Define Δ_i, Γ_i as follows:

$$\Delta_1 = \{<\}, \Delta_2 = \{\leq\}, \Delta_3 = \{=\}, \Delta_4 = \{\underline{\|}\}$$

$$\Gamma_1 = \emptyset, \Gamma_2 = \{R_1\}, \Gamma_3 = \{R_2\}, \Gamma_4 = \{R_3, R_4\}$$

We demonstrate that Δ_i is not 2-independent of Γ_i by giving examples that are not satisfiable although all subinstances containing at most two relations from Δ_i are satisfiable.

1. $x_1 < x_2 < x_3 < x_1$
2. $x_1 \leq x_2 \leq x_3 \leq x_1, x_1\ R_1\ x_2$
3. $x_1 = x_2 = x_3 = x_4, x_1\ R_2\ x_4$
4. $x_1 \underline{\|} x_2 \underline{\|} x_3 \underline{\|} x_4, x_1\ R_3\ x_2\ R_3\ x_3\ R_3\ x_4, x_1\ R_4\ x_4$

Since Δ_i is not 2-independent of Γ_i, NP-hardness follows from Theorem 10. $\square$

4.3 Maximality

We are now ready to prove that $\mathcal{S}_A, \ldots, \mathcal{S}_D$ are the only maximal tractable sets of the disjunctive point algebra for partially ordered time. To reduce the number of NP-completeness results needed in the classification, we will employ a modified closure operator $C_{\text{po}}(\cdot)$ which is defined with the aid of the standard operators converse, intersection and composition together with a number of equivalences. These equivalences and their properties are stated in the next lemma. The straightforward proof is only sketched.

Lemma 23 Let $R_i \in \mathcal{PA}$ and $\Gamma_j \subseteq \mathcal{PA}$. The following problems are equivalent up to polynomial-time reductions.

1. $\textsc{Psat}_{po}(\Gamma_1 \cup \{< \vee R_1\})$ and $\textsc{Psat}_{po}(\Gamma_1 \cup \{< \vee R_1,\ R_1 \vee R_1\})$;
2. $\textsc{Psat}_{po}(\Gamma_1 \cup \{= \vee R_1)$ and $\textsc{Psat}_{po}(\Gamma_1 \cup \{= \vee R_1,\ R_1 \vee R_1\})$ when $<, <>, \neq$ or $< \|$ is in Γ_1.
3. $\textsc{Psat}_{po}(\Gamma_1 \cup \{R_1 \vee R_3, R_2 \vee R_3\})$ and $\textsc{Psat}_{po}(\Gamma_1 \cup \{R_1 \vee R_3, R_2 \vee R_3, (R_1 \oplus R_2) \vee R_3\})$ when $\oplus \in \{\cap, \circ\}$;

Proof sketch: The proofs of these equivalence are fairly similar so we only prove the first one. It is sufficient to note that $aR_1b \vee cR_1d$ is equivalent to

$$\{e < f \vee aR_1b,\ f < e \vee cR_1d\}$$

where e, f are fresh variables. $\square$

If Γ is a set of relations, we define $C_{\mathrm{po}}(\Gamma)$ to be the least set X such that

1. $\Gamma \subseteq X$;
2. X is closed under converse, intersection and composition of point relations; and
3. X is closed under the rules in the previous lemma.

By using Lemma 23, the $\overset{\times}{\vee}$-closure property and the properties of converse, intersection and composition, it is a routine verification to show that if $\textsc{Psat}_{po}(\Gamma)$ is NP-complete, then $\textsc{Psat}_{po}(C_{\mathrm{po}}(\Gamma))$ is also NP-complete.

We can now prove our classification theorem.

Theorem 24 $\mathcal{S}_A, \mathcal{S}_B, \mathcal{S}_C$ and $\mathcal{S}_D$ are the only maximal tractable subclasses of $\textsc{Psat}_{po}$.

Proof: Suppose to the contrary that there exists another maximal tractable algebra $\mathcal{T}$. It follows that there exists $\gamma_A, \cdots, \gamma_D$ in $\mathcal{T}$ such that $\gamma_A \in \overline{\mathcal{S}}_A, \cdots, \gamma_D \in \overline{\mathcal{S}}_D$ by Lemma 6. Note that there exists only a finite number of possible values for $\gamma_A, \ldots, \gamma_D$.

To prove the result, a machine-assisted case analysis of the following form was performed: each admissible choice of $\gamma_A, \ldots, \gamma_D$ was generated and $X = C_{\mathrm{po}}(\{\gamma_{\mathrm{A}}, \ldots, \gamma_{\mathrm{D}}\})$ was computed. Each such set X was examined and it was found that at least one of the NP-complete sets of Lemma 22 was a subset of X. Thus, $\textsc{Psat}_{po}(\mathcal{T})$ is NP-complete and the theorem follows. $\square$

4.4 Partial orders and spatial reasoning

This section contains an example where we use the complexity results for point algebras in order to show computational properties of a spatial reasoning problem. The basic idea is to exhibit a connection between partial orders and convex sets in the plane. When this is done, the translation of complexity results is fairly trivial.

Let E^n denote the n-dimensional Euclidean space. Given a set $S \subseteq E^n$, we say that S is *convex* iff for every two points $x, y \in S$ and every $0 \leq \theta \leq 1$, $\theta \cdot x + (1-\theta) \cdot y \in S$. Define the set Θ such that $x \in \Lambda$ if and only if x satisfies the following properties:

1. x is a convex set in the plane, i.e. a convex subset of E^2; and

2. the area of x is strictly larger than 0.

We require the area of objects to be > 0 in order to rule out degenerate (i.e. lines and points) objects. In the Λ-*problem*, we consider objects taken from Λ related by the following basic relations:

1. $x \subset y$ iff $x \subset y$ (i.e. x is a strict subset of y);

2. $x \supset y$ iff $x \supset y$;

3. $x = y$ iff $x = y$;

4. $x \| y$ iff $x \not\subset y$, $x \not\supset y$ and $x \neq y$.

The possible relations between intervals are the 2^4 possible unions of these basic relations. The satisfiability problem is that of deciding whether there exists elements in Λ that can be assigned to the variables such that all of the relations are satisfied. We will now prove that there is a close connection between the Λ-problem and reasoning about partially-ordered time. More precisely, we prove the following theorem.

Theorem 25 Let Π be an instance of the Λ-problem. Then, Π is satisfiable iff the following $\text{PSAT}_{po}(\mathcal{PA})$ problem Π' is satisfiable: for each variable x in Π, introduce a variable x and for each constraint xRy in Π, impose the constraint $x \bigcup_{r \in R} \sigma(r) y$ where $\sigma(\subset) = (<)$, $\sigma(\supset) = (>)$, $\sigma(=) = (=)$ and $\sigma(\|) = (\|)$.

This theorem combined with the results in the previous sections tells us, for instance, that the Λ-problem has three maximal tractable subclasses (corresponding to Γ_A, Γ_B and Δ_D) and four maximal disjunctive tractable subclasses (corresponding to $\mathcal{S}_A, \ldots, \mathcal{S}_D$).

We need a couple of definitions and lemmata to prove this theorem. Given two points a, b in the plane, the *(Euclidean) distance* between them is

$$d(a, b) = \sqrt{(a_1 - b_1)^2 + (a_2 - b_2)^2}.$$

The distance between a and b equals 0 iff $a = b$. Let $C \subseteq E^2$ denote the unit circle, i.e. $\{(x, y) \mid x^2 + y^2 = 1\}$. Given a set $S \subseteq E^n$, $\text{cnv}(S)$ denotes the *convex hull* of S, i.e.

$$\text{cnv}(S) = \{x\theta + y(1-\theta) \mid x, y \in S \text{ and } 0 \leq \theta \leq 1\}.$$

Note that if S is a subset of C and $|S| > 2$, then the area of $\text{cnv}(S)$ is > 0.

Lemma 26 If $x, a, b \in C$ and $x = \theta a + (1-\theta)b$ for some $0 \leq \theta \leq 1$, then $x = a$ or $x = b$.

Proof: If $\theta = 0$, then $x = b$ and if $\theta = 1$, then $x = a$. Assume that $0 < \theta < 1$. If $z = (z_1, z_2) \in C$, then $z_1^2 + z_2^2 = 1$ by the definition of the unit circle. Hence,

$$
\begin{array}{ll}
 & x_1^2 + x_2^2 = 1 \\
\Leftrightarrow & (\theta a_1 + (1-\theta)b_1)^2 + (\theta a_2 + (1-\theta)b_2)^2 = 1 \\
\Leftrightarrow & \theta^2 \underbrace{(a_1^2 + a_2^2)}_{1} + (1-\theta)^2 \underbrace{(b_1^2 + b_2^2)}_{1} + 2\theta(1-\theta)(a_1b_1 + a_2b_2) = 1 \\
\Leftrightarrow & \theta^2 + (1-\theta)^2 + 2\theta(1-\theta)(a_1b_1 + a_2b_2) = 1 \\
\Leftrightarrow & 2\theta(\theta - 1) + 2\theta(1-\theta)(a_1b_1 + a_2b_2) = 0 \\
\Leftrightarrow & 2\theta(1-\theta) = 2\theta(1-\theta)(a_1b_1 + a_2b_2) \\
\Leftrightarrow & 1 = a_1b_1 + a_2b_2 \text{ (since } 0 < \theta < 1 \text{ implies } 2\theta(1-\theta) \neq 0).
\end{array}
$$

We calculate the Euclidean distance between a and b:

$$
\begin{array}{rcl}
d(a,b) & = & \sqrt{(a_1 - b_1)^2 + (a_2 - b_2)^2} \\
 & = & \sqrt{a_1^2 + b_1^2 - 2a_1b_1 + a_2^2 + b_2^2 - 2a_2b_2} \\
 & = & \sqrt{\underbrace{a_1^2 + a_2^2}_{1} + \underbrace{b_1^2 + b_2^2}_{1} - \underbrace{2(a_1b_1 + a_2b_2)}_{2}} \\
 & = & 0.
\end{array}
$$

Consequently, $a = b$. Furthermore, $x = \theta a + (1-\theta)b$ so $x = a = b$. $\square$

Lemma 27 Let P, Q be arbitrary subsets of C and $r \in \{\subset, \supset, =, ||\}$. If PrQ, then $\text{cnv}(P)\ r\ \text{cnv}(Q)$.

Proof: The lemma holds trivially if $r = (=)$. Assume now that $r = (\subset)$. Arbitrarily choose $x \in \text{cnv}(P)$. Clearly, $x = \theta a + (1-\theta)b$ for some $a, b \in P$ and $0 \leq \theta \leq 1$. It follows immediately that $x \in \text{cnv}(Q)$ since $P \subset Q$ and $a, b \in Q$.

Now, choose $x \in Q - P$. We show that $x \notin \text{cnv}(P)$ and thus prove the lemma. Assume to the contrary that $x = \theta a + (1-\theta)b$ for some $a, b \in P$ and $0 \leq \theta \leq 1$. Since $x, a, b \in C$, we can apply Lemma 26 and conclude that $x = a$ or $x = b$. Consequently, $x \in P$ which leads to a contradiction. The case $r = \supset$ is analogous.

$P||Q$ implies the existence of x, z such that:

- $x \in P$, $x \notin Q$;
- $y \notin P$, $y \in Q$;

We show that $x \notin \text{cnv}(Q)$ and $y \notin \text{cnv}(P)$ which establishes the result. Assume to the contrary that $x \in \text{cnv}(Q)$. Then, there exists $a, b \in Q$ such that $x = \theta a + (1-\theta)b$ for some $0 \leq \theta \leq 1$. Now, $x, a, b \in C$ and $x = a$ or $x = b$ by Lemma 26. Thus, $x \in Q$ which contradicts our initial assumptions. Showing that $y \notin \text{cnv}(P)$ is analogous. $\square$

Proof: (of Theorem 25)

only-if: The structure $(\Lambda, \subseteq)$ is a partial order so Π' is satisfiable if Π is satisfiable.

if: Assume Π' contains the variables $X = \{x_1, \ldots, x_n\}$ and that $f : X \rightarrow (T, \leq)$ is a model of Π'. We assume without loss of generality that $T = \{2, \ldots, n+1\}$. Construct the function $f' : X \rightarrow 2^T$ as follows:

$$f'(x_i) = \{0, 1, f(x_i)\} \cup \{j \in T \mid j \text{ is below } f(x_i) \text{ in } (T, \leq)\}.$$

It is easy to realize that if $f(x) <_T f(y)$ implies $f'(x) \subset f'(y)$, $f(x) \|_T f(y)$ implies $f'(x) \| f'(y)$ and so forth. Arbitrarily choose an injective function $g : \{0, \ldots, n+1\} \rightarrow C$. Construct an interpretation I of Π as follows: for $1 \leq i \leq k$,

$$I(x_i) = \mathrm{cnv}(\{g(t) \mid t \in f'(x_i)\}).$$

Since $I(x_i)$ is the convex hull of three or more points of the unit circle, it is a convex set in the plane with area > 0. It remains to show that if $x_i r x_j \in \Pi$, then x_i and x_j are related by r under the interpretation I. However, this follows immediately from the definition of function σ and Lemma 27. $\square$

5 Partial Orders with Bounded Dimension

We begin this section by considering a simple scenario: a number of sensors $S_1, \ldots, S_k$ are observing a system and they transmit time-stamped data to a central computer. The system is highly asynchronous so each sensor runs a different clock and these clocks are not synchronized. Assume a certain event A occurs within the system. Each sensor S_i observes A and gives it a time-stamp t_i. Since there is no global notion of time, the best we can do is to assign A the time-stamp $(t_1, \ldots, t_k)$, i.e. a vector containing the time-stamps of the different sensors.

Now, assume we have two events A, A' with time-stamps $(t_1, \ldots, t_k)$ and $(t'_1, \ldots, t'_k)$, respectively. If $t_i = t'_i$ for $1 \leq i \leq k$, we know that A and A' occurred simultaneously; if $t_i \leq t'_i$, $1 \leq i \leq k$, and there exists a j such that $t_j < t'_j$, we know that A occurred strictly before A' and so on.

To reason about such systems, we can construct time point algebras where the possible time points are real vectors containing k elements (i.e. the set of time points equals the k-dimensional Euclidean space) and the basic relations are defined as follows: for time points $x = (x_1, \ldots, x_k), y = (y_1, \ldots, y_k)$:

1. $x < y$ iff $\forall i : x_i \leq y_i$ and $\exists j : x_j < y_j$
2. $x > y$ iff $\forall i : x_i \geq y_i$ and $\exists j : x_j > y_j$
3. $x = y$ iff $\forall i : x_i = y_i$
4. $x \| y$ otherwise

This time model is closely related to certain restricted partial orders and this connection gives us the opportunity to study the complexity of such point algebras. We need a few definitions:

Let $P = (T, \leq)$ be a partial order. A *linear extension* of P is a partial order $P' = (T, \leq')$ such that $\leq \subseteq \leq'$ and for every $x, y \in T$, either $(x, y) \in \leq'$ or $(y, x) \in \leq'$.

Given a set of partial orders $\mathcal{T} = \{(T, \leq_1), \ldots, (T, \leq_n)\}$, the *intersection* $\bigcap \mathcal{T} = (T, \bigcap_{i=1}^{n} \leq_i)$ is also a partial order. The *dimension* $d(P)$ of a partial order is the least k such that there exists a set $\mathcal{T}$ of total orders over T satisfying

1. $|\mathcal{T}| = k$; and
2. $P = \bigcap \mathcal{T}$.

Note that every finite partial order has well-defined dimension. The dimension of a partial order $P = (T, \leq)$ can be given a well-known geometric interpretation [34]: let π be a mapping from T to distinct points of the d-dimensional Euclidean space E^d. Let $P(\pi) = (T, \leq')$ denote the partial order defined by: $(x, y) \in \leq'$ if and only if each coordinate of $\pi(x)$ is less than or equal the corresponding coordinate of $\pi(y)$. The dimension of P is then the minimum d for which there exists such a mapping π with $P(\pi) = P$.

Hence, the time model presented above coincides with partially-ordered time where the set of partial orders have bounded dimension. Note that the ordinary partially-ordered time model corresponds to a system containing an infinite number of unsynchronized clocks. Let po/k denote the class of partial orders with dimension bounded by k and note that the set of total orders equals the set $po/1$. Thus, $\textsc{Psat}_{po/1}(\mathcal{PA})$ is the satisfiability problem for the totally ordered time point algebra.

We continue our study of point algebras with bounded dimension by demonstrating the hardness of reasoning about satisfiability even for basic relations. This is followed by a complete classification of the tractable sets of relations and, finally, we also give a complete classification for the case when disjunctions are allowed.

5.1 Basic relations

We will now show that reasoning about partial orders of bounded dimensions is NP-hard even when restricted to the two basic relations $<, \|$. Apart from demonstrating the difficulty of reasoning about satisfiability over this domain this also lays the foundation for the total classifications in the next two sections.

Theorem 28 Let $X = \{<, \|\}$. Then, $\textsc{Psat}_{po/k}(X)$ is NP-complete whenever $k \geq 2$.

The fact that we cannot solve the satisfiability problem for the basic relations implies, among other things, that straightforward backtracking algorithms for $\textsc{Psat}_{po/k}(\mathcal{PA})$, $k \geq 2$, will not work. Such algorithms successively refines the relations in the given instance until only basic relation remains and this modified problem is solved by using some tractable method (such as enforcing path consistency).

The proof of Theorem 28 is relatively easy when $k \geq 3$.

Lemma 29 $\text{PSAT}_{po/k}(X)$ is NP-complete when $k \geq 3$.

Proof: Yannakakis [41] has proved that deciding whether a given partial order has dimension k or not is an NP-complete problem for every fixed $k \geq 3$. Thus, given a partial order $P = (T, \leq)$, we can in polynomial time construct an instance of $\text{PSAT}_{po/k}(X)$ that is satisfiable iff the given partial order has dimension k. For each $t \in T$, introduce a variable t. If two points t and t' are related under $<$, add the constraint $t < t'$; if they are incomparable, add the constraint $t \| t'$. $\Box$

When $k = 2$, we use a connection between intervals on the real line and partial orders of dimension 2.

Theorem 30 (Baker, Fishburn and Roberts [6]) A partial order $(T, \leq)$ has dimension 2 iff for each $t \in T$, there exists an interval I_t on the real line such that $t \leq t' \Leftrightarrow I_t \subseteq I_{t'}$.

The problem can now be proved by using Allen's algebra. Allen's interval algebra [1] is based on the notion of *relations between intervals.* An interval x is represented as a tuple (x^-, x^+) of real numbers with $x^- < x^+$, denoting the left and right endpoints of the interval, respectively. The possible relations between intervals are the 2^{13} possible unions of 13 *basic interval relations*, which are shown in Table 3. The problem of *satisfiability* ($\mathcal{A}$-SAT(S)) of a set of interval variables with relations from S between them is that of deciding whether there exists an assignment of intervals on the real line for the interval variables, such that all of the relations between the intervals are satisfied. The complexity of this problem has been determined for all choices of S [28].

Theorem 31 $\text{PSAT}_{po/2}(X)$ is NP-complete.

Proof: The problem $\mathcal{A}\text{-SAT}(\{(\mathsf{dsf}), (\mathsf{pp}^{-1}\mathsf{mm}^{-1}\mathsf{oo}^{-1})\})$ is NP-complete [28]. Given an instance of this problem, it can trivially be transformed to an instance of $\text{PSAT}_{po/2}(X)$:

- for each constraint $x(\mathsf{dsf})y$, add the constraint $x < y$; and
- for each constraint $x(\mathsf{pp}^{-1}\mathsf{mm}^{-1}\mathsf{oo}^{-1})y$, add the constraint $x \| y$.

By Theorem 30, the resulting instance is satisfiable iff the given instance of Allen's algebra is satisfiable. $\Box$

Basic relation		Example	Endpoints
x precedes y	p	xxx	$x^+ < y^-$
y preceded by x	p^{-1}	yyy	
x meets y	m	xxxx	$x^+ = y^-$
y met by x	m^{-1}	yyyy	
x overlaps y	o	xxxx	$x^- < y^- < x^+$,
y overl. by x	o^{-1}	yyyy	$x^+ < y^+$
x during y	d	xxx	$x^- > y^-$,
y includes x	d^{-1}	yyyyyyy	$x^+ < y^+$
x starts y	s	xxx	$x^- = y^-$,
y started by x	s^{-1}	yyyyyyy	$x^+ < y^+$
x finishes y	f	xxx	$x^+ = y^+$,
y finished by x	f^{-1}	yyyyyyy	$x^- > y^-$
x equals y	$\equiv$	xxxx	$x^- = y^-$,
		yyyy	$x^+ = y^+$

Table 3: The thirteen basic relations. The endpoint relations $x^- < x^+$ and $y^- < y^+$ that are valid for all relations have been omitted

5.2 Total classification of the point algebra

We continue by giving a total classification of the point algebra. We prove that the following sets of relations are the only maximal tractable sets of relations for $\mathrm{PSAT}_{po/k}$ for $k \geq 2$.

1. $\mathcal{A}_1 = \{<, \leq, <>, < \| >, <=>, =\}$
2. $\mathcal{A}_2 = \{\|, < \|, < \| >, =, \underline{\|}, \leq \|\}$
3. $\mathcal{A}_3 = \{=, \leq, <=>, \underline{\|}, \leq \|\}$

Corollary 18 implies the tractability of $\mathrm{PSAT}_{po/k}(\mathcal{A}_1)$. Tractability for the two other sets of relations is trivial since $\mathcal{A}_2 = \Delta_{\|} \cup \{=\}$, $\mathcal{A}_3 = \Delta_{=}$ and $\Delta_{\|}, \Delta_{=}$ both have the GS property. We need a few auxiliary results to show that these are the only maximal tractable sets of relations.

Lemma 32 Let $P = (V, <_P)$ be a partial order of dimension $k \geq 2$. Then the following partial orders are also of dimension k.

1. $P' = (P \cup \{s\}, <_P \cup \{(y, s) | y <_P x\})$ for any $x \in V$.
2. $P'' = (P \cup \{s\}, <_P \cup \{(s, y) | x <_P y\})$ for any $x \in V$.

Proof: We only prove the first case since the second case can be proven analogously. Let $<_1, \ldots, <_k$ be k linear extensions of $<_P$ such that $<_P = <_1 \cap \cdots \cap <_k$. Furthermore, let $<_1'$ be $<_1 \cup \{(a, s), (s, b) | a, b \in P, a \leq_1 x, x <_1 b\}$

and let $<_i'$ be $<_i \cup \{(a,s) | a \in P\}$ when $i > 1$. Obviously, $<_1', <_i'$ are total orders and $<_{P'} = <_1' \cap \cdots \cap <_k'$ which demonstrates that P' has the dimension k. $\square$

Intuitively, the previous lemma states that we can always add a new maximal (case 1) or minimal (case 2) element connected to exactly one $x \in P$ (not counting the transitive relations) without increasing the dimension of the partial order.

Lemma 33 $\text{P}\small{\text{SAT}}_{po/k}(\{<, \underline{\|}\})$ is NP-complete for $k \geq 2$.

Proof: We prove this by a reduction from the previously proven NP-hard problem $\text{P}\small{\text{SAT}}_{po/k}(\{<, \|\})$. Let Π be a $\text{P}\small{\text{SAT}}_{po/k}(\{<, \|\})$ instance and let Π' be the result of replacing each occurrence $x \| y$ of $\|$ in Π with the constraints $x < s, y < t, s \underline{\|} t, s \underline{\|} y, t \underline{\|} x$ where s, t are two fresh variables. We prove that Π is satisfiable iff Π' is satisfiable. For the if-direction assume that Π' is satisfiable and has a model M. Then it must hold that $x \| y$ under M so M is also a model of Π.

For the only-if case assume that Π is satisfiable and has a model M. Let M' be the result of extending M with the two points s, t such that $x < s, y < t$. Obviously this interpretation satisfies all the constraints in Π' and Lemma 32 proves that this extension does not increase the dimension of the partial order. Thus, M' is a model of Π'. $\square$

Lemma 34 $\text{P}\small{\text{SAT}}_{po/k}(\{\leq, \|\})$ is NP-complete for $k \geq 2$.

Proof sketch: A reduction similar to that in the previous lemma from the $\text{P}\small{\text{SAT}}_{po/k}(\{<, \|\})$ case can be made by replacing constraints $x < y$ with the constraints $x \leq y, x \| s, s \leq y$ where s is a fresh variable. $\square$

Lemma 35 $\text{P}\small{\text{SAT}}_{po/k}(\{\|, <>\})$ is NP-complete for $k > 1$.

Proof: In Section 4.2, it was proven that $\text{P}\small{\text{SAT}}_{po}(\{\|, <>\})$ is NP-hard by a reduction from the NP-complete problem BETWEENNESS. The reduction creates an $\text{P}\small{\text{SAT}}_{po}(\{\|, <>\})$ instance Π for each BETWEENNESS instance (A, T) and proves that Π is satisfiable iff (A, T) is satisfiable. Obviously, even if we regard Π as an $\text{P}\small{\text{SAT}}_{po/k}$ problem instance the only-if part still holds.

For the if part we prove that $\mathcal{P}_{|A|}$ is of dimension two and hence, that the proof of the if part in Section 4.2 holds also for all other finite dimensions. Let g, h be the two injective mappings from the points of $\mathcal{P}_{|A|}$ onto $\{1, \ldots, 3|A|\}$ such that:

$$g(x_i) = |A| + 1 - i, h(z_i) = 3|A| + 1 - i$$
$$g(y_i) = |A| + 2i - 1, h(y_i) = 2i$$
$$g(z_i) = |A| + 2i, h(x_i) = 2i - 1$$

These two mapping give us two total orders $<_1 = \{(s,t) | g(s) < g(t)\}$, $<_2 = \{(s,t) | h(s) < h(t)\}$. A visualization of $\mathcal{P}_4$ and the corresponding linear extensions can be found in Figure 4. We see that $<_1 \cap <_2 = \mathcal{P}_{|A|}$ and thus $\mathcal{P}_{|A|}$ is of dimension two. Hence, the reduction from BETWEENNESS holds also for partial orders of fixed dimension ≥ 2. $\square$

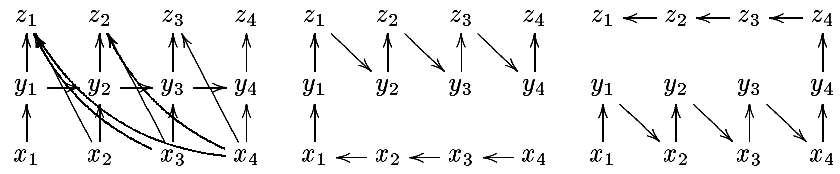

Figure 4: The partial order $\mathcal{P}_4$ and corresponding linearisations

Lemma 36 $\text{PSAT}_{po/k}(\{<=>, \|\})$ is NP-complete for $k \geq 2$.

Proof: We make a reduction from $\text{PSAT}_{po/k}(\{<>, \|\})$ by replacing each constraint $x <> y$ with the constraints $x <=> s, s\|y, x <=> y$ where s is a fresh variable. Since $x = y$ contradicts $x <=> s, s\|y$ we see that the original problem instance is satisfiable if this problem instance is. For the only-if direction assume that the original problem instance has a model $M = (f, (T, <_M))$. We construct an interpretation $M' = (f, (T \cup \{s\}, <'_M))$ where:

$$<_{M'} = \begin{cases} (T \cup \{s\}, <_M \cup \{(z,s) | z <_M x\}) & \text{if } x <_M y \\ (T \cup \{s\}, <_M \cup \{(s,z) | x <_M z\}) & \text{otherwise} \end{cases}$$

Obviously M' satisfies all the constraints in the original problem instance and it also satisfies all the new constraints introduced in the reduction. From Lemma 32 we see that the dimension of M' is the same as that for M. Hence, the reduced problem is satisfiable iff the original problem is. □

We are now ready to prove the main theorem of this subsection. By proving that there exists no other maximal tractable set of relations than $\mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_3$ we have given a total classification for the point algebra for partial orders of bounded dimensions.

Theorem 37 $\mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_3$ are the only maximal tractable subclasses of $\text{PSAT}_{po/k}$, for $k \geq 2$.

Proof: Assume that there exists some other maximal tractable set of relations $\mathcal{A}$. From Lemma 6, it follows that the following must hold:

1. $\exists r_1 \in \overline{\mathcal{A}}_1 = \{\|, <\|, \underline{\|}, \leq \|\}$ and $r_1 \in A$
2. $\exists r_2 \in \overline{\mathcal{A}}_2 = \{<, <>, \leq, <=>\}$ and $r_2 \in A$
3. $\exists r_3 \in \overline{\mathcal{A}}_3 = \{<, <>, <\|, <\|>, \|\}$ and $r_3 \in A$

If r_1 is $\|$ or $<\|$, then (regardless of the choice of r_2) $\text{PSAT}_{po/k}(\{r_1, r_2\})$ is NP-complete according to the previous lemmata. Otherwise, r_1 is $\underline{\|}$ or $\leq \|$. If r_2 is $<$ or $<>$, then $\text{PSAT}_{po/k}(\{r_1, r_2\})$ is also NP-complete according to the previous lemmata. If r_2 is $\leq$ or $<=>$ we either have that r_3 is $\|$ which makes $\text{PSAT}_{po/k}(\mathcal{A})$ NP-complete for the same reason as when $r_1 = \|$, or we have that r_3 is $<\|$ or $<\|>$. We see that $r_2 \cap r_3$ is $<$ or $<>$ and $\text{PSAT}_{po/k}(\mathcal{A})$ is NP-complete since $\text{PSAT}_{po/k}(\{r_1, r_2 \cap r_3\})$ is NP-complete. □

5.3 Disjunctions

In this subsection we extend the results of the previous classification to allow disjunctions. We prove that the following sets of relations are the only maximal tractable subclasses of $\textsc{Psat}_{po/k}$ extended with disjunctions:

1. $\mathcal{B}_1 = \Gamma_1 \stackrel{\times}{\vee} \Delta_1^*$ where $\Delta_1 = \{<>, <\,\|\,>, <=>\}$ and $\Gamma_1 = \Delta_1 \cup \{<, \leq, =\}$
2. $\mathcal{B}_2 = \Gamma_2 \stackrel{\times}{\vee} \Delta_2^*$ where $\Delta_2 = \{\|, <\,\|, <\,\|\,>, \underline{\|}, \leq\,\|\}$ and $\Gamma_2 = \Delta_2 \cup \{=\}$
3. $\mathcal{B}_3 = \Delta_3^*$ where $\Delta_3 = \{=, \leq, <=>, \underline{\|}, \leq\,\|\}$

Tractability of $\textsc{Psat}_{po/k}(\mathcal{B}_1)$ follows from tractability of $\textsc{Psat}_{po/1}(\mathcal{X}_1)$ by using Corollary 18. Furthermore, since Δ_2 and Δ_3 have the GS property, tractability of $\textsc{Psat}_{po/k}(\mathcal{B}_2)$ and $\textsc{Psat}_{po/k}(\mathcal{B}_3)$ follows immediately. We continue with a few NP-hardness results needed for the classification result.

Lemma 38 $\textsc{Psat}_{po/k}(\{r_1, r_2\})$, $k \geq 2$, is an NP-complete problem whenever $r_1 \in \{<, <>, <\,\|, \neq, \|\}$ and $r_2 \in (\Gamma_1 - \Delta_1)^2 - (\Gamma_1 - \Delta_1)$.

Proof: First, choose γ_1, γ_2 such that $r_2 = \gamma_1 \vee \gamma_2$. In order to reduce the number of cases needed to be examined we note that $\gamma_1 \vee \gamma_2 = \gamma_2 \vee \gamma_1$. Furthermore if γ_1 is $\leq$ then $\textsc{Psat}_{po/k}(\{r_1, r_2\})$ can trivially be reduced to $\textsc{Psat}_{po/k}(\{r_1, = \vee \gamma_2\})$. It is therefore sufficient to only examine the cases where r_2 is one of $< \vee <, = \vee =$ or $< \vee =$.

For the first and the second case, it has already been proven that r_2 is not 2-independent of r_1 in Lemma 22 for the case when $k = \infty$. Obviously, the very same counterexamples holds also for partial orders of bounded dimensions so $\textsc{Psat}_{po/k}(\{r_1, r_2\})$ is NP-hard.

For the third case, we make a reduction from the case when r_2 is $= \vee =$ by replacing each constraint $c : x = y \vee z = w$ with the two constraints $c_1 : x = y \vee t < s$ and $c_2 : s < t \vee z = w$ where s, t are two fresh variables. Note that every model satisfying c_1 and c_2 also satisfies c. We prove the other direction by noting that any model $M = (f, (T, <_M))$ satisfying c can be extended to a model $M' = (f \cup f', (T \cup \{u, v\}, <_M \cup \{(u, v), (u, a), (v, a) | a \in T\}))$ where $f' = \{(s, u), (t, v)\}$ if $f(x) = f(y)$ and $f' = \{(s, v), (t, u)\}$ otherwise. Obviously, M' satisfies all the constraints and Lemma 32 implies that M' has the same dimension as M. Thus, this problem instance is satisfiable iff the original problem instance is. $\square$

Together with the total classification of the previous subsection this lemma can now be used to prove NP-hardness of every set of relation not included in $\mathcal{B}_1, \mathcal{B}_2$ or $\mathcal{B}_3$.

Theorem 39 $\mathcal{B}_1, \mathcal{B}_2, \mathcal{B}_3$ are the only maximal tractable sets of relations for $\textsc{Psat}_{po/k}$ extended with disjunctions for all $k \geq 2$.

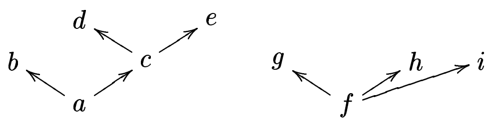

Figure 5: A small subset of **br**.

Proof: Let $\mathcal{B}$ be a set of relations which is not a subset of one of $\mathcal{B}_1, \mathcal{B}_2, \mathcal{B}_3$. It follows from Lemma 6 that there exists $x_1 \in \overline{\mathcal{B}}_1, x_2 \in \overline{\mathcal{B}}_2, x_3 \in \overline{\mathcal{B}}_3$ such that $x_1, x_2, x_3 \in \mathcal{B}$. We will examine every possible choice of x_1, x_2, x_3 and demonstrate that deciding satisfiability for $\mathcal{B}$ is NP-hard. We have two cases.

First, assume that $x_1 \notin \mathcal{PA} - \Gamma_1$. We must then have that $x_1 \in (\Gamma_1 - \Delta_1)^2 - (\Gamma_1 - \Delta_1)$ and from Lemma 38 we see that deciding satisfiability for $\{x_1, x_3\}$ is NP-hard. Furthermore, if $x_2 \notin \mathcal{PA} - \Gamma_2$ then x_2 is $= \vee =$ and Lemma 38 can be applied again which implies NP-hardness of $\{x_2, x_3\}$.

Thus we must have $x_1 \in \mathcal{PA} - \Gamma_1, x_2 \in \mathcal{PA} - \Gamma_2, x_3 \in \mathcal{PA} - \Delta_3$. From Theorem 37 we see that deciding satisfiability for $\{x_1, x_2, x_3\}$ is NP-hard. $\square$

6 Branching Time

We will now turn our attention to the point algebra for branching time. This point algebra restricts the class of partial orders $(T, \leq)$ to those that satisfy the *branching* condition:

$$\forall x, y, z \in T : \text{if } x \| y \text{ and } y < z \text{ then } x \| z.$$

The set of partial orders satisfying this condition is denoted *br*. Given a partial order $(T, \leq) \in br$, we note that two incomparable points cannot have an upper bound. This tree-like structure motivates the name 'branching'. We also note that $po/1 \subset br \subset po$ but br and po/k are incomparable when $1 < k < \infty$.

In the following, we will view *br* as a forest containing all finite trees infinitely many times; we denote this set **br**. We choose this definition since it simplifies the proofs and does not change the basic model of time. Consequently, we define the basic relations as follows: given arbitrary points x, y in **br**,

1. $x < y$ iff x is below y in **br**.
2. $x > y$ iff y is below x in **br**.
3. $x = y$ iff x and y are the same point.
4. $x \| y$ iff x, y belong to different branches or trees.

Example 6.1 In the subset of **br** (depicted in Figure 5) some of the relations holding between points are: $a < c$, $b \| c$, $b \| e$, $a < d$ and $c \| f$.

	$<$	$>$	$\parallel$	$=$
$<$	$<$	$<=>$	$<\parallel$	$<$
$>$	$\top$	$>$	$\parallel$	$>$
$\parallel$	$\parallel$	$>\parallel$	$\top$	$\parallel$
$=$	$<$	$>$	$\parallel$	$=$

Table 4: The composition table for the point algebra for branching time.

A point $n \in \mathbf{br}$ related to some point in the image of f by $<$ or $>$ but not itself in the image is said to be a *redundant* point. If there exists no redundant points for f, ie. if f satisfies the following:

$$f(V) = \{a \in \mathbf{br} \mid \exists x \in V : a \leq f(x) \vee f(x) \leq a\}$$

we say that f is a *non-redundant* model of Π. Note that if Π has a model, then Π also has a non-redundant model since it is always possible to remove an arbitrary point from a tree and preserve the relations between remaining points.

Example 6.2 Let Π be a problem instance over five variables $\{x_0, \ldots, x_4\}$ and the constraints: $x_0 < x_3, x_2 \neq x_3, x_4 \| x_3, x_2 \leq x_4, x_4 \leq x_2, x_1 \| x_4, x_0 <> x_1$ and $x_0 <> x_4$. Furthermore let, f_1, f_2 and f_3 be the following interpretations:

$$f_1(x_0) = a, f_2(x_0) = f, f_3(x_0) = a$$

$$f_1(x_1) = b, f_2(x_1) = g, f_3(x_1) = g$$

$$f_1(x_2) = d, f_2(x_2) = h, f_3(x_2) = d$$

$$f_1(x_3) = e, f_2(x_3) = i, f_3(x_3) = e$$

$$f_1(x_4) = d, f_2(x_4) = h, f_3(x_4) = d$$

where $a, \ldots, i$ are the point in $\mathbf{br}$ given in Figure 5. We have that f_1 and f_2 are models of Π but f_3 is not since the $x_0 <> x_1$ constraint is not satisfied by f_3. Furthermore f_2 is a non-redundant model while f_1 is redundant since c is connected to the nodes of f_1 but not itself part of the image of f_1.

6.1 Tractability results

We will now show that the sets of relations defined below (where $\Psi_A, \ldots, \Omega_E$ are defined in Table 5) are tractable.

$$\mathcal{T}_A = \Psi_A, \;\; \mathcal{T}_B = \Psi_B \stackrel{\times}{\vee} \Omega_B^*, \;\; \mathcal{T}_C = \Omega_C^*, \;\; \mathcal{T}_D = \Psi_D \stackrel{\times}{\vee} \Omega_D^*, \;\; \mathcal{T}_E = \Psi_E \stackrel{\times}{\vee} \Omega_E^*.$$

We note that all basic relations are included in $\mathcal{T}_A$ and $\mathcal{T}_E$. The subclass $\mathcal{T}_A$ is a bit unusual since it cannot be extended by disjunctions at all—in the case

	Ψ_A	Ψ_B	Ω_B	Ω_C	Ψ_D	Ω_D	Ψ_E	Ω_E
$<$	$\bullet$	$\bullet$					$\bullet$	
$\leq$	$\bullet$	$\bullet$		$\bullet$			$\bullet$	
$<>$	$\bullet$	$\bullet$	$\bullet$				$\bullet$	
$<=>$	$\bullet$	$\bullet$	$\bullet$	$\bullet$			$\bullet$	
$\|$	$\bullet$				$\bullet$	$\bullet$	$\bullet$	
$\underline{\|}$	$\bullet$			$\bullet$	$\bullet$	$\bullet$		
$=$	$\bullet$	$\bullet$		$\bullet$	$\bullet$		$\bullet$	
$\neq$	$\bullet$	$\bullet$	$\bullet$		$\bullet$	$\bullet$	$\bullet$	$\bullet$
$<\|$	$\bullet$				$\bullet$	$\bullet$	$\bullet$	
$\leq\|$	$\bullet$			$\bullet$	$\bullet$	$\bullet$		

Table 5: Tractable classes in branching time.

of partially-ordered time, every maximal tractable subclass can be extended by disjunctions. The class $\mathcal{T}_B$ always has a model that is a total order by Corollary 18. Furthermore, tractability of $\mathcal{T}_C$ and $\mathcal{T}_D$ is trivial since Ω_C and Ω_D has the GS property. These sets of relations will be proven to be the unique maximal sets of tractable relations for PSAT_{br} in Sections 6.2-6.3.

The tractability of Ψ_A follows from Hirsch's [24] $O(n^5)$ algorithm. We present a considerably faster $O(n^{3.326})$ algorithm for this problem in Section 6.4. We will now proceed to the four other tractable sets of relations.

Theorem 40 The satisfiability problem for $\mathcal{T}_B, \ldots, \mathcal{T}_E$ is tractable.

Proof: We note that every problem instance of $\text{PSAT}_{po}(\mathcal{T}_B)$ and $\text{PSAT}_{po}(\mathcal{T}_C)$ is satisfiable iff it has a totally ordered model (by Corollary 18) or a one-point model, respectively. Hence, we can solve $\text{PSAT}_{\text{br}}(\mathcal{T}_B)$ and $\text{PSAT}_{\text{br}}(\mathcal{T}_C)$ problem instances as problem instances of the point algebra for totally ordered time as described in Section 3.

The following two steps is all that is needed for an algorithm determining satisfiability for problem instances Π of $\text{PSAT}_{\text{br}}(\Psi_D)$.

1. For each constraint $c = (=, x, y)$, contract x and y.
2. If there exists a constraint (R, x, x) such that $= \not\subseteq R$ then reject, else accept.

If the algorithm rejects an instance, it is clearly not satisfiable. Otherwise, it is satisfiable since all remaining variables can be mapped to root nodes in disjoint trees. By an analysis of the algorithm it is obvious that Ω_D is independent of Ψ_D and, hence, $\Psi_D \stackrel{\times}{\vee} \Omega_D^*$ is tractable.

Finally, a tractable algorithm for Ψ'_E can be found in Section 6.4. By analyzing this algorithm in the same way as we analyzed algorithm A in the proof of Lemma 19, we see that $\neq$ is independent of Ψ'_E and hence, $\Psi'_E \stackrel{\times}{\vee} \{\neq\}$ is tractable.

Finally, it is easy to show that $\Psi_E = C(\Psi'_E)$ so $\mathcal{T}_E$ is tractable by Theorem 2. $\square$

It is important to note that the automatic method for proving independence suggested by Broxvall *et al.* [10] cannot be used for proving the previous theorem. In order to use this method, the satisfiability problem of the underlying set of relations must be decided by path consistency. However, Hirsch [24] has proved that path consistency does not decide $\textsc{Psat}_{\mathrm{br}}(\{<>, <\|\})$.

6.2 Intractability results

This section contains the NP-completeness results which are needed for proving the classification result.

Theorem 41 $\textsc{Psat}_{\mathrm{br}}$is NP-hard for the following sets of relations:

1. $\{< \vee <\}$
2. $\{= \vee =, R_1\}$ where $=$ is not a subset of R_1
3. $\{\| \vee \|, <=>\}$
4. $\{\neq \vee \neq, \underline{\|}, <>\}$
5. $\{\underline{\|} \vee \underline{\|}, <, <=>\}$
6. $\{R_2 \vee R_2, \|\}$ where R_2 is $<>$ or $<=>$
7. $\{<> \vee <=>, \|\}$

Proof: NP-hardness of the first six sets of relations will be proven in a fashion similar to that in Lemma 13. We begin by defining Δ_i, Γ_i as follows:

$$\Delta_1 = \{<\}, \Delta_2 = \{=\}, \Delta_3 = \{\|\}, \Delta_4 = \{\neq\}, \Delta_5 = \{\underline{\|}\}, \Delta_6 = \{R_2\}$$

$$\Gamma_1 = \emptyset, \Gamma_2 = \{R_1\}, \Gamma_3 = \{<=>\}, \Gamma_4 = \{\underline{\|}, <>\}, \Gamma_5 = \{<, <=>\}, \Gamma_6 = \{\|\}$$

Now consider the six problem instances below.

1. $x_1 < x_2 < x_3 < x_1$
2. $x_1 = x_2 = x_3 = x_4, x_1 \; R_1 \; x_4$
3. $x_1 <=> x_2 <=> x_3 <=> x_4 <=> x_5 <=> x_6 <=> x_1, x_1 \| x_4, x_2 \| x_5, x_3 \| x_6$
4. $x_1 <> x_2 <> x_3 <> x_4 <> x_5 <> x_6 <> x_1, x_1 \underline{\|} x_4, x_2 \underline{\|} x_5, x_3 \underline{\|} x_6, x_1 \neq x_4, x_2 \neq x_5, x_3 \neq x_6$
5. $x_1 <=> x_2 < x_3 <=> x_4 < x_5 <=> x_6 < x_1, x_1 \underline{\|} x_2, x3 \underline{\|} x_4, x_5 \underline{\|} x_6$
6. $x_1 \; R_2 \; x_2 \; R_2 \; x_3 \; R_2 \; x_4 \; R_2 \; x_5 \; R_2 \; x_6 \; R_2 \; x_1, x_1 \| x_4, x_2 \| x_5, x_3 \| x_6$

By using the algorithm in Figure 6, it can be shown that these instances are unsatisfiable even though all subinstances of them containing at most two constraints from Δ_i are satisfiable. Thus, Δ_i is not 2-independent of Γ_i and Theorem 10 implies that deciding satisfiability for the first six sets of relations is NP-hard.

The NP-hardness proof for the final set of relations cannot be derived in the same way. We therefore make a reduction from the NP-hard problem $\text{PSAT}_{\text{br}}\{<> \vee <>, \|\}$. Let C be a set of constraints over these relations. We construct a new set of constraints C' by replacing every occurrence of $c : a <> b \vee c <> d$ with the constraints $\{c_1, c_2\} \cup S$:

$$c_1 : a <> b \vee x_1 <=> x_2, \;\; c_2 : c <> d \vee x_2 <=> x_3$$

$$S = \{x_3 <=> x_4 <=> x_5 <=> x_6 <=> x_1, x_1 \| x_4, x_2 \| x_5, x_3 \| x_6\}$$

where $x_1, \ldots, x_6$ are fresh variables. By using the algorithm in Figure 6, it is fairly easy to see that the constraints $\{x_1 <=> x_2\} \cup S$ and $\{x_2 <=> x_3\} \cup S$ are both satisfiable but not $\{x_1 <=> x_2, x_2 <=> x_3\} \cup S$. It follows that any model of C easily can be extended to a model of C'. We also see that in any model of C' we must either have that $a <> b$ or $c <> d$ is satisfied. Thus a model of C' is also a model of C. $\square$

6.3 Maximality

We will now demonstrate that the five sets of relations proven tractable in Section 6.1 are the only maximal tractable sets. To do so, we need a number of equivalence results.

Lemma 42 Let $R_i \in \mathcal{PA}$ and $\Gamma_j \subseteq \mathcal{PA}$. The following problems are equivalent up to polynomial-time reductions.

1. $\text{PSAT}_{\text{br}}(\{< \vee R_1\} \cup \Gamma_1\})$ and $\text{PSAT}_{\text{br}}(\{(<) \vee R_1, R_1 \vee R_1\} \cup \Gamma_1)$.
2. $\text{PSAT}_{\text{br}}(\{= \vee R_1, R_2\} \cup \Gamma_1)$ and $\text{PSAT}_{\text{br}}(\{= \vee R_1, R_2, R_1 \vee R_1\} \cup \Gamma_1)$ where $= \not\subseteq R_2$.
3. $\text{PSAT}_{\text{br}}(\{\| \vee R_1, R_2\} \cup \Gamma_1)$ and $\text{PSAT}_{\text{br}}(\{\| \vee R_1, R_2, R_1 \vee R_1\} \cup \Gamma_1)$ where R_2 is $(<>)$ or $(<=>)$.
4. $\text{PSAT}_{\text{br}}(\{\underline{\|} \vee R_1, < \|, R_2\} \cup \Gamma_1)$ and $\text{PSAT}_{\text{br}}(\{\underline{\|} \vee R_1, < \|, R_2, R_1 \vee R_1\})$ where R_2 is $(<>)$ or $(<=>)$.
5. $\text{PSAT}_{\text{br}}(\{\neq \vee R_1, \|, R_1\} \cup \Gamma_1)$ and $\text{PSAT}_{\text{br}}(\{\neq \vee \neq, \neq \vee R_1, \|, R_1\} \cup \Gamma_1)$ whenever R_1 is the relation $(<>)$ or $(<=>)$;
6. $\text{PSAT}_{\text{br}}(\{R_1 \vee R_2, R_1 \vee R_3\} \cup \Gamma_1)$, $\text{PSAT}_{\text{br}}(\{R_1 \vee R_2, R_1 \vee R_3, R_1 \vee (R_2 \oplus R_3)\} \cup \Gamma_1)$ when $\oplus \in \{\circ, \cap\}$.

Proof: The first two cases were proved in Lemma 23 for partially-ordered time and it is easy to show that they hold also for branching time. For the third case consider the following set of relations:

$$a\ R'\ b,\ b\ R'\ c,\ c\ R'\ d,\ x\ R\ y \vee a||c,\ z\ R\ w \vee b||d$$

and note that at most one of $a||c$ and $b||d$ can hold. Hence, $x\ R\ y \vee z\ R\ w$ must hold but without further restricting the choices of x, y, z, w. We use this in order to make a polynomial reduction from $\text{PSAT}_{\text{br}}(\{|| \vee R, R', R \vee R\} \cup \Gamma)$ to $\text{PSAT}_{\text{br}}(\{|| \vee R, R'\} \cup \Gamma)$. Let Π be an arbitrary problem instance of the first set of relations and replace each $R \vee R$ constraint of Π with the constraints above where. Clearly, the resulting set of constraints is an instance of $\text{PSAT}_{\text{br}}(\{|| \vee R, R'\} \cup \Gamma)$ that is satisfiable iff the original instance is satisfiable. Correctness of the next two points follows by similar reasoning. The final point follows from the following two equivalences.

- $x\ \gamma_1\ y \vee z\ (\gamma_2 \cap \gamma_3)\ w$ is equivalent to $x\ \gamma_1\ y \vee z\ \gamma_2\ w$ and $x\ \gamma_1\ y \vee z\ \gamma_3\ w$
- $x\ \gamma_1\ y \vee z\ (\gamma_2 \circ \gamma_3)\ w$ is equivalent to $x\ \gamma_1\ y \vee z\ \gamma_2\ t$ and $x\ \gamma_1\ y \vee t\ \gamma_3\ w$ where t is a fresh variable.

□

If Γ is a set of relations, we define $C_{\text{br}}(\Gamma)$ as the least set X such that

1. $\Gamma \subseteq X$;
2. X is closed under converse, intersection and composition on point relations; and
3. X is closed under the equivalences in the previous lemma.

It is a routine verification to show that if $\text{PSAT}_{po}(\Gamma)$ is tractable, then the problem $\text{PSAT}_{po}(C_{\text{po}}(\Gamma))$ is also tractable. We can now prove the main result by using a computer assisted case analysis as in the proof of Theorem 24.

Theorem 43 $\mathcal{T}_A, \mathcal{T}_B, \mathcal{T}_C, \mathcal{T}_D$ and $\mathcal{T}_E$ are the only maximal tractable disjunctive subclasses of PSAT_{br}.

6.4 Algorithm for Ψ_A

We will describe an alternative algorithm for Ψ_A in this section; the algorithm can be found in Figure 6. It runs in $O(nM(n))$ time, where $M(n)$ denotes the time complexity of multiplying two $n \times n$ matrices. Using the $O(n^{2.376})$ algorithm for matrix multiplication proposed by Coppersmith and Winograd [14], this algorithm is a significant improvement over the $O(n^5)$ algorithm given by Hirsch [24].

The algorithm works by first partitioning the problem instance into sets of variables which can be mapped to disjoint trees, ie. all constraints between the partitions include the relation $||$. Next, the algorithm tries to identify a variable

Algorithm $C(\Pi)$
Input: Problem instance $\Pi = (V, C)$ of Ψ_A
Let G be the undirected graph $(V, \emptyset)$
for each constraint $xRy \in C$ such that $\| \not\subseteq R$ **do**
  add the edge $\{x, y\}$ to G
Partition G into components $V_1, \ldots, V_k$
Partition C into $C_1, \ldots, C_k$ such that:
$C_i = \{xRy \in C | x, y \in V_i\}$
for each component V_i **do**
  Let G_i be the directed graph $(V_i, \emptyset)$
  for each $c = xRy \in C_i$ such that $< \not\subseteq R$ **do**
    add the (directed) edge (x, y) to G_i
  Let M be the adjacency matrix of the transitive
  and reflexive closure of G_i
  $N \leftarrow$ new empty matrix indexed by V_i
  for each $xRy \in C_i$ such that $= \not\subseteq R$ **do**
    $N[x, y] \leftarrow 1$
  $P \leftarrow M \cdot N$
  $root \leftarrow \perp$
  for each $x \in V_i$ **do**
    if $\forall y \in V_i : P[x, y] = 0$ or $M[x, y] = 0$ **then**
      $root \leftarrow x$
  if $root \neq \perp$ **then**
    $V_i' \leftarrow V_i - \{c | M[root, c] = 1\}$
    $C_i' \leftarrow \{xRy \in C_i \mid x, y \in V_i'\}$
    if BRANCH$((V_i', C_i'))$ rejects **then** **return false**
  else **return false**
return true

Figure 6: Algorithm for determining satisfiability for Ψ_A

root which can be mapped to the root node for each partition. Note that there exists exactly one such node for each satisfiable partition since each partition maps to a distinct tree.

Let x_1 be a candidate variable to be mapped to the root node in a satisfiable problem instance. If there exists a chain of constraints $x_1\ R_1\ x_2\ \cdots\ x_{n-1}\ R_n\ x_n$ such that $< \not\subseteq R_i$ we must also map $x_2, \ldots, x_n$ to the root node. Existence of the constraints $x_1\ R_1\ x_2\ \cdots\ x_{n-1}\ R_n\ x_n$ for each pair of variables x_1, x_n is checked by looking at the adjacency matrix M of the transitive closure for the graph G_i constructed by the algorithm for every partition V_i. If $M[x, y] = 1$ and x is mapped to the root node then so must y.

Thus, for x to be mapped to the root node there must not exist a pair of variables which both are identified with x and for which we have a constraint not allowing the equality relation, ie. there must not exist variables y, z and a constraint $c = yRz$ such that $= \not\subseteq R$, $M[x, y] = 1$ and $M[x, z] = 1$. Conveniently, existence of z and c such that $= \not\subseteq R$ and $M[x,z] = 1$ can be computed by matrix multiplication. This is done in step 18, where $P[x,y] = 1$ iff there exists such a variable z and constraint c. After identifying a variable *root* which can be mapped to the root node, every variable which must be identified with *root* is removed from the partition and the algorithm is recursively called for this new problem instance.

Before proving correctness of the algorithm we recall the definition of matrix multiplication.

$$C = A \cdot B \Leftrightarrow C[i,j] = \sum_{k=1}^{n} A[i,k] \cdot B[k,j] \tag{1}$$

Lemma 44 If algorithm C rejects a problem instance then it is not satisfiable.

Proof: Assume to the contrary that there exists a satisfiable problem instance Π which is rejected by the algorithm. Then there exists a component V_i of a subset of the original problem instance such that the algorithm rejects in line 27. Note that rejection on line 26 only occurs if the algorithm rejected on line 27 after some recursions. Let Π' denote the problem instance (V_i, C_i) where V_i, C_i is the variables and constraints making the algorithm reject on line 27. Trivially, the algorithm rejects also Π'. Note that Π' is a subinstance of Π and, hence satisfiable.

Since Π' is satisfiable there exist a non-redundant model f of Π'. Assume that there exists more than one minimal point in the image of Π' under f and let $f(x)$ and $f(y)$ denote two such points. Note that minimal points in **br** are the roots of distinct trees. Since the graph G constructed by the algorithm for Π' contains only one component then there exists a path of constraints $x\ R_1\ x_1\ \cdots\ x_n\ R_{n+1}\ y$ such that $\| \not\subseteq R_1, \ldots, R_{n+1}$. Since $f(x)$ and $f(y)$ are the roots of different trees we obviously have some x_i, x_{i+1} whose images lies in different trees thus violating the constraint $x_i\ R_{i+1}\ x_{i+1}$. Contradiction, and we have only one minimal point which must be the root of all other points in the image in Π'.

Let x denote a variable in Π' such that $f(x)$ is the minimal point. Since the algorithm rejects we know that there exists some $y \in V_i$ such that $P[x,y] \geq 1$ and $M[x,y] = 1$. >From the definition of matrix multiplication (1) follows the equations (2) and (3) which obviously are equivalent.

$$P[x,y] = \sum_{z \in V_i} M[x,z] \cdot N[z,y] \geq 1 \tag{2}$$

$$\exists z \in V_i : M[x,z] = N[z,y] = 1 \tag{3}$$

Hence, we have $z \in V_i$, $c = zRy \in C_i$ such that $M[x,z] = 1$ and $= \not\subseteq R$. Since $M[x,y] = 1$, we have a chain of constraints $x\ R_1\ y_1\ R_2\ \cdots\ R_{n+1}\ y$ such that $< \not\subseteq R_i$. Note that $f(x) \leq f(y_i)$ and hence, $f(x) = f(y_1)$ and $f(y_1) \leq f(y_i)$ which gives $f(x) = f(y_1) = f(y_2)$ and $f(y_2) \leq f(y_i)$ etc. This leads to $f(x) = f(y_i) = f(y)$. Analogously, $f(x) = f(z_i) = f(z)$. Contradiction, since $f(z) = f(x) = f(y)$ violates the constraint c. Thus, Π' and Π are not satisfiable. $\square$

Having proven that the algorithm rejects only unsatisfiable instances we will now demonstrate that the algorithm correctly identifies the satisfiable problem instances.

Lemma 45 Algorithm C only accepts satisfiable problem instances.

Proof: We prove the result by induction over the number of variables in the given problem instance. Obviously, all problem instances accepted by the algorithm containing zero variables are satisfiable. Assume that acceptance of the algorithm implies satisfiability for all problem instances of size n or less. Let Π be a problem instance of size $n+1$ which is accepted by the algorithm. We construct an interpretation f of the problem instance as follows:

1. For each component i, let f_i' denote a model of (V_i', C_i). Note that $|V_i'| < |V|$ since at least one variable is removed on line 24. Existence of f_i' follows from the induction hypothesis.

2. Let V_i' be an arbitrary component and T_i the subset of **br** that f_i' maps V_i' to; i.e. **br** restricted to $\{x \in \mathbf{br} \mid x = f_i'(v) \text{ and } v \in V_i'\}$. Consider another subtree U_i of **br** that is isomorphic to T_i with one exception; it contains a fresh root point t_i (which is below all other points). Modify f_i' to map V' to $U_i - \{t_i\}$ instead.

   We define an interpretation f_i of (V_i, C_i) as follows:

$$f_i(x) = \begin{cases} t_i & \text{iff } M[root, x] = 1 \\ f_i'(x) & \text{otherwise} \end{cases}$$

3. Define f as the union of the (disjoint) interpretations f_i.

We continue by demonstrating that each constraint $C = xRy$ is satisfied by f. Assume $x \in V_i$ and $y \in V_j$. We have two cases.

1. If $i = j$ there are four cases to analyze. If $M[root, x] = M[root, y] = 0$ then C is included in (V_i, C_i) and hence, satisfied by f_i',f_i and thus f. When $M[root, x] = M[root, y] = 1$ we have $P[root, y] = 0$ which implies $N[x, y] = 0$. Hence, we have $= \subseteq R$ and thus, C is satisfied by f_i and f. Finally, when $M[root, x] = 1$ and $M[root, y] = 0$ we have $< \subseteq R$ since there otherwise would exist a path from $root$ to y in the graph G_i constructed by the algorithm. Hence, f_i and f satisfies C. The case $M[root, x] = 0$ and $M[root, y] = 1$ can be shown analogously.

2. Assume $i \neq j$. Since the algorithm has partitioned x and y into different components we know that $\| \subseteq R$ and c is satisfied by f.

□

By demonstrating that the algorithm runs in polynomial time we conclude that the satisfiability problem for $\mathcal{T}_A$ is tractable. This is done in the following theorem.

Theorem 46 Algorithm C runs in $O(nM(n))$ time.

Proof: We begin by noting that the initial graph partitioning can be performed in $O(n^2)$ steps by applying a standard graph partitioning algorithm, cf. Baase [5]. Also, the transitive closure of G_i on line 13 can be computed in the same time as the boolean matrix multiplication on line 18 [5]. Hence, steps 10-26 can all be performed within $O(M(n_i))$ time. The function $f(n)$ is an upper bound of the total running time of algorithm C:

$$f(n) \leq c_1 n^2 + \sum_{i=1}^{k} (c_2 M(n_i) + f(n_i - 1))$$

for some constants c_1, c_2 and where n_i denotes the size of component V_i. Note that we have $n_1 + \ldots + n_k = n$. Furthermore, for all $c \geq 0$ we have that $n_1^c + \cdots + n_k^c \leq (n_1 + \cdots + n_k)^c$. Hence, $\sum_{i=1}^{k} c_2 M(n_i) \leq c_2 M(\sum_{i=1}^{k} n_i) = c_2 M(n)$. This gives us:

$$f(n) \leq (c_1 + c_2)M(n) + \sum_{i=1}^{k} f(n_i - 1)$$

and it can be shown that $f(n)$ is a polynomial with positive coefficients by induction over n. Consequently, the worst case appears when $k = 1$, that is,

$$f(n) \leq (c_1 + c_2)M(n) + f(n-1) = \sum_{i=1}^{n} (c_1 + c_2)M(i).$$

Given a polynomial p that is everywhere greater than zero, it holds that $\sum_{i=1}^{n} p(i) \leq n \cdot p(n)$. Hence,

$$f(n) \leq \sum_{i=1}^{n} (c_1 + c_2)M(i) \leq (c_1 + c_2)nM(n)$$

□

6.5 Algorithm for Ψ_E

This section contains the final part of the proof of Theorem 40; that is, we present a polynomial-time algorithm for $\textsc{Psat}_{\mathrm{br}}(\Psi'_E)$ (where $\Psi'_E = \{\leq, \|, \neq\}$).

A variable n is said to be *superminimal* in a problem instance Π iff n is a $(\leq)$-minimal variable of Π and there exists no x such that $n \| x$. A *semipath* between two variables n_1, n_2 in an instance Π is a set of variables $a_1, \ldots, a_k, b_1, \ldots, b_k$ such that:

$$n_1 \leq^* a_1 \geq^* b_1 \leq^* \ldots \leq^* a_k \geq^* b_k \leq^* n_2$$

Note that if two variables n_1, n_2 are part of the same $(\leq)$-component then there exists a semipath between n_1, n_2. Before we can prove the main result of this section, we need several lemmata.

Lemma 47 If $a \leq^* b$ and $b <=> c$, then $a <=> c$.

Proof: Follows from the composition table for branching time. □

In the point algebra of branching time we have an important property which helps us prove correctness of algorithm D. Only superminimal variables in a problem instance Π may be mapped to a minimal point in any model of Π, which the following lemma proves.

Lemma 48 Assume that Π is a problem instance containing variables n_1, n_2 such that $n_1 \| n_2$ and there exists a semipath between n_1, n_2, the image of n_1 cannot be a minimal node in any model of Π.

Proof: Assume M is a model of Π such that $M(n_1)$ is a minimal node. Since there exists a semipath between n_1, n_2 there exists $a_1, \ldots, a_k$,$b_1, \ldots, b_k$ such that:

$$n_1 \leq^* a_1 \geq^* b_1 \leq^* \ldots \leq^* a_k \geq^* b_k \leq^* n_2$$

Lemma 47 gives $n_1 <=> b_1$, $M(n_1)$ minimal gives $n_1 \leq b_1$. Note that $n_1 \leq b_1 \leq^* a_2$ and $b_2 \leq^* a_2$, Lemma 47 gives $n_1 <=> b_2$ thus $n_1 \leq b_2$. Repeat for each b in $b_2, \cdots, b_k$ and we have $n_1 \leq b_k \leq^* n_2$. Hence $n_1 \leq^* n_2$ and $n_1 \| n_2$, contradiction. Thus there exists no model M such that $M(n_1)$ is a minimal node. □

Lemma 49 Let Π be a problem instance of $\text{PSAT}_{br}(\Psi'_E)$ such that Π is $(\leq)$-acyclic and f is a non-redundant model of Π. Then there exists at least one minimal point p in **br** which is the image of a $(\leq)$-minimal variable in Π.

Proof: Arbitrarily choose a variable $v \in \text{Var}(\Pi)$ and let T be the unique subtree of **br** such that $f(v) \in T$. Let p be a minimal point in T.

Let $N = n_1, \cdots, n_k$ be the variables in Π that are mapped to p by f. We know that M is nonredundant so $N \neq \emptyset$. Assume now that no member of N is $(\leq)$-minimal. Π is acyclic so there exists a $(\leq)$-minimal variable n' such that $n' \leq^* n_1 \in N$. The $(\leq)$-minimality of n' implies that $n' \notin N$ and $f(n') \neq p$. This leads immediately to a contradiction since $f(n') < f(n_1) = p$ and p is minimal in $(T, \leq)$. Consequently, there exists a $(\leq)$-minimal variable in N that is mapped to p. □

Now, we can prove that algorithm D (see Figure 7) correctly solves $\text{PSAT}_{br}(\Psi'_E)$. Given a problem instance Π and a variable v, we write $\Pi - \{n\}$ to denote the problem instance $(V - \{v\}, C')$ where

$$C' = \{xry \in C \mid x \neq v \text{ and } y \neq v\}.$$

Lemma 50 Algorithm D correctly solves the $\text{PSAT}_{br}(\Psi'_E)$ problem.

Proof: We prove correctness of the algorithm by induction over the number of variables in the instance. The algorithm accepts all instances Π containing zero variables and such instances are trivially satisfiable.

Assume the algorithm correctly solves the problem for every instance containing k variables. Let Π be an arbitrary instance containing $k+1$ variables. We consider four cases:

1. (lines 5-7) There exists two variables n_1, n_2 such that $n_1 \leq^* n_2, n_2 \leq^* n_1$. If $n_1 \neq n_2$ in Π, then there can be no model of Π and the algorithm rejects. Otherwise, by Lemma 15 Π is satisfiable iff contract(Π, n_1, n_2) is satisfiable and the algorithm is correct by the induction hypothesis.

2. (lines 10-11) There exists at least two $(\leq)$-components $C_1, \cdots, C_n$ of Π. If the algorithm rejects some C_i, then C_i cannot have a model by the induction hypothesis. Since C_i is a subinstance of Π, there exist no model of Π.

   We show that if C_i is satisfiable for every i then Π is satisfiable. For each $i \leq n$, let f_i denote a model of C_i and assume without loss of generality that the ranges of these models are different subtrees of **br**.

   Create a new interpretation f by taking the union of $f_1, \ldots, f_n$. Note that the only possible relation r between two distinct variables $n_i \in C_i, n_j \in C_j$ can be $\neq$ or $\|$. Obviously, $f(n_i) r f(n_j)$ holds since n_i and n_j are mapped to different trees. Thus, it follows that f is an interpretation of Π since every f_i is an interpretation of C_i.

3. (lines 12-13) There exists only one $(\leq)$-component of Π which has a superminimal variable n. If the algorithm rejects, then $\Pi - \{n\}$ is unsatisfiable by the induction hypothesis and Π cannot have a model.

   If $\Pi - \{n\}$ is satisfiable, then Π is satisfiable: let f' denote a model of $\Pi - \{n\}$ and T_i the subset of **br** that f' maps the variables in Π to. Consider another subtree U_i of **br** that is isomorphic to T_i with one exception; it contains a fresh root point t (which is below all other points). Modify f' to map the variables to $U_i - \{t\}$ instead. Now, simply define an interpretation of Π as follows: $f(n) = t$ and $f(v) = f'(v)$ for all other variables.

   Clearly, f is an interpretation of $\Pi - \{n\}$ since the only added point t is a minimal point. Assume $nrv \in \Pi$. If r is $\neq$, then the relation holds trivially. Otherwise r is $\leq$ since the superminimality of n prohibits r from being either $\|$ or $\geq$. Hence, the relation holds since $f(n)$ is a minimal point.

1 **Algorithm** $D(\Pi)$
2 **Input** An instance Π of $\textsc{Psat}_{br}(\Psi'_E)$
3 **if** Π is empty **then**
4   **return true**
5 **elsif** $\exists n_1, n_2 \in \text{Var}(\Pi)$ such that $n_1 \leq^* n_2, n_2 \leq^* n_1 \in \Pi$ **then**
6   **if** $n_1 \neq n_2 \in \Pi$ **then return false**
7   **else return** $D(\text{contract}(\Pi, n_1, n_2))$
8 **else**
9   Identify the $(\leq)$-components $C_1, \cdots, C_k$ of Π.
10   **if** $k > 1$ **then**
11   **return** $D(C_1) \wedge D(C_2) \wedge \cdots \wedge D(C_k)$
12   **elsif** $\exists n \in \text{Var}(\Pi)$ such that n is superminimal **then**
13   **return** $D(\Pi - \{n\})$
14   **else**
15   **return false**
16   **end if**
17 **end if**

Figure 7: The algorithm for solving $\textsc{Psat}_{br}(\Psi'_E)$

4. (line 15) There exists only one $(\leq)$-component of Π and Π contains no superminimal variable. Assume Π is satisfiable. Then, there exists a non-redundant model M of Π. By Lemma 49, M has a minimal point p which is the image of some $(\leq)$-minimal variable n. Since n is a minimal variable but not a superminimal variable there exists an x such that $n \| x$. Since Π only has one component there exists a $(\leq)$-semipath between n and x. By Lemma 48 the image p of n cannot be a minimal variable which leads to a contradiction. Hence, there cannot exist a model of Π.

$\square$

Lemma 51 $\textsc{Psat}_{br}(\Psi'_E)$ is tractable.

Proof: The correctness of algorithm D follows from Lemma 50. To show that D runs in polynomial time, let $p(n)$ be a polynomial that is an upper bound for the time complexity of which steps 5 and 9 can be performed. The time complexity of the algorithm is bounded from above by the solution to the following recursive equation:

$$\begin{aligned} f(1) &= 1 \\ f(n) &= \max \begin{cases} p(n) + f(n-1) \\ p(n) + \sum_{i=1}^{k} f(c_i) & : k \geq 2, c_i > 0, \sum_{i=1}^{k} c_i = n \end{cases} \end{aligned}$$

The case $f(n) = p(n) + f(n-1)$ occurs when the algorithm recurse at line 13 and the case $f(n) = \sum f(c_i)$ occurs when the algorithm recurse at line 11 (where c_i denotes the size of component C_i). Proving that $f(n) \leq c \cdot np(n)$ for some constant c is analogous to the proof of Theorem 46. $\square$

7 Concluding Remarks

We have studied the computational complexity of a number of temporal reasoning problems in different time models such as totally-ordered, partially-ordered and branching time. Primarily, we have considered the satisfiability problem for point algebras and point algebras extended with disjunctions—for these problems, we have identified all tractable subclasses. We have also considered a number of other issues such as a new time model suitable for reasoning about systems with a bounded number of unsynchronized clocks, improved algorithms for branching time and examples of how our complexity results can be transferred to other domains.

We have only considered qualitative relations between time points in this article. For some applications this is satisfactory but for others we must have the ability to reason about metric time. Reasoning about metric time in the totally-ordered model of time is straightforward from a conceptual point of view—each time point is typically viewed as a real or rational number. Furthermore, the linear-programming approach by Jonsson & Bäckström [25] and Koubarakis [27] offers a straightforward method for reasoning about metric information; for instance, these methods can be used for extending Allen's algebra with metric constraints. The satisfiability problem for the resulting formalism is still NP-complete so the extension does not increases the computational complexity. It should also be noted that the ORD-Horn class [33] is still tractable after such an extension.

Adding metric constraints to partially-ordered models of time is more difficult. In this case, it is not obvious what is meant by metric constraints. The approach taken in Section 5 can fairly easily be extended to a reasonable notion of metric time; each time point is a k-vector of real numbers. We can extend this idea to ordinary partially-ordered time by considering a time point to be a function from the natural numbers to the real numbers, i.e. an infinite vector of reals. Whether this idea is to be considered reasonable or not may of course be discussed. An intuitively appealing definition of metric time in the branching time model seems to be even less obvious.

Algorithmic methods for identifying tractable disjunctive constraints have been presented earlier in the literature [10]. The methods used in this paper complements this idea by providing automatic methods for proving intractability of disjunctive constraints (as an example, consider the proofs in Section 6.2). By demonstrating that sets of relations of the form $\Gamma \cup \Delta^2$ are tractable only if Δ is 2-independent of Γ, we have an automatic method for identifying NP-hard sets of relations; one can show that Δ is not 2-independent of Γ by exhaustively generating problem instances and testing for satisfiability. Even though this is not necessarily a complete method, note that many of the NP-hard sets of relations used in this article were identified using this method. Furthermore, for the domains where path-consistency decides consistency for the tractable fragments (such as $\textsc{Psat}_{po}$ and $\textsc{Psat}_{to}$), tractable disjunctive extensions can also be identified using an automatic method [10]. Together, these two methods are probably useful tools for classifying new domains. This raises the question under which conditions the methods are sufficient to provide a complete classification.

Acknowledgments

We would like to thank Víctor Dalmau and Debasis Mitra for providing helpful comments on earlier drafts of this paper. This research has been supported by the ECSEL graduate student program, the *Swedish Research Council for the Engineering Sciences* (TFR) under grant 97-301 and by the *Swedish Research Council* (VR) under grant 221-2000-361.

References

[1] J. F. Allen, Maintaining knowledge about temporal intervals, Communications of the ACM 26 (11) (1984) 832–843.

[2] F. Anger, On Lamport's interprocessor communication model, ACM Transactions on Programming Languages Systems 11 (3) (1989) 404–417.

[3] F. Anger, R. Rodriguez, The lattice structure of temporal interval relations, Journal of Applied Intelligence 6 (1) (1996) 29–38.

[4] F. Anger, D. Mitra, R. Rodriguez, Temporal constraint networks in nonlinear time, Technical report, ECAI Workshop on Temporal and Spatial Reasoning, 1998.

[5] S. Baase, Computer Algorithms: Introduction to Design and Analysis, Addison Wesley, Reading, MA, 2nd edition, 1988.

[6] K. R. Baker, P. C. Fishburn, F. S. Roberts, Partial orders of dimension 2, Networks 2 (1972) 11–28.

[7] M. Broxvall, The point algebra for branching time revisited, in: Proceedings Joint German/Austrian Conference on Artificial Intelligence (KI/ÖGAI-2001), Vienna, Austria, 2001, pp. 106-121.

[8] M. Broxvall, P. Jonsson, Towards a complete classification of tractability in point algebras for nonlinear time, in: Proceedings 5th International Conference on Principles and Practice of Constraint Programming (CP-1999), Alexandria, VA, USA, 1999, pp. 129–143.

[9] M. Broxvall, P. Jonsson, Disjunctive temporal reasoning in partially ordered time structures, in: Proceedings 17th National Conference on Artificial Intelligence (AAAI-2000), Austin, TX, USA, 2000, pp. 464–469.

[10] M. Broxvall, P. Jonsson, J. Renz, Disjunctions, independence, refinements, Artificial Intelligence, to appear.

[11] M. Broxvall, P. Jonsson, J. Renz, Refinements and independence: a simple method for identifying tractable disjunctive constraints, in: Proceedings 6th International Conference on Principles and Practice of Constraint Programming (CP-2000), Singapore, 2000, pp. 114–127.

[12] E. Clarke, E. Emerson, A. Sistla, Automatic verification of finite-state concurrent systems using temporal logic specifications, ACM Transactions on Programming Languages Systems 8 (2) (1986) 244–263.

[13] D. Cohen, P. Jeavons, P. Jonsson, M. Koubarakis, Building tractable disjunctive constraints, Journal of the ACM 47 (5) (2001) 826–853.

[14] D. Coppersmith, S. Winograd, Matrix multiplication via arithmetic progressions, Symbolic Computation 9 (3) (1990) 251–280.

[15] T. Dean and M. Boddy, Reasoning about partially ordered events, Artificial Intelligence 36 (1988) 375–399.

[16] T. Drakengren and M. Bjäreland, Expressive reasoning about action in nondeterministic polynomial time, in: Proceedings 16th International Joint Conference on Artificial Intelligence (IJCAI-99), Stockholm, Sweden, 1999, pp. 166–171.

[17] T. Drakengren, M. Bjäreland, Reasoning about action in polynomial time, Artificial Intelligence 115 (1) (1999) 1–24.

[18] I. Düntsch, H. Wang, S. McCloskey, Relations algebras in qualitative spatial reasoning, Fundamenta Informaticae 39 (3) (1999) 229–248.

[19] E. A. Emerson, J. Y. Halpern, "Sometimes" and "not never" revisited: On branching versus linear temporal logic, Journal of the ACM 33 (1) (1986) 151–178.

[20] E. A. Emerson, J. Srinivasan, Branching time temporal logic, in: J. de Bakker, W.-P. de Roever, G. Rozenberg, (Eds.), Linear Time, Branching Time and Partial Order in Logics and Models for Concurrency, Springer-Verlag, New York, 1989, pp. 123–172.

[21] E. A. Emerson, J. Srinivasan, Branching time temporal logic, in: J. de Bakker, W.-P. de Roever, and G. Rozenberg, (Eds.), Linear Time, Branching Time and Partial Order in Logics and Models for Concurrency, Springer-Verlag, 1989, pp. 123–172.

[22] M. Garey, D. Johnson, Computers and Intractability: A Guide to the Theory of NP-Completeness, Freeman, New York, 1979.

[23] M. C. Golumbic, R. Shamir, Complexity and algorithms for reasoning about time: A graph-theoretic approach, Journal of the ACM 40 (5) (1993) 1108–1133.

[24] R. Hirsch, Expressive power and complexity in algebraic logic, Journal of Logic and Computation 7 (3) (1997) 309–351.

[25] P. Jonsson, C. Bäckström, A unifying approach to temporal constraint reasoning, Artificial Intelligence 102 (1) (1998) 143–155.

[26] P. Jonsson, T. Drakengren, C. Bäckström, Computational complexity of relating time points with intervals, Artificial Intelligence, 109 (1–2) (1999) 273–295.

[27] M. Koubarakis, Tractable disjunctions of linear constraints: basic results and applications to temporal reasoning, Theoretical Computer Science 266 (1–2) (2001) 311-339.

[28] A. Krokhin, P. Jeavons, P. Jonsson, Reasoning about temporal relations: the tractable subalgebras of Allen's interval algebra, Technical report PRG-RR-01-12, Computing Laboratory, Oxford University, 2001. Available from `web.comlab.ox.ac.uk/oucl/publications/tr/rr-01-12.html`.

[29] L. Lamport, Time, clocks and the ordering of events in a distributed system, Communications of the ACM 21 (7) (1978) 558–565.

[30] L. Lamport, The mutual exclusion problem: Part I—a theory of interprocess communication, Journal of the ACM 33 (2) (1986) 313–326.

[31] D. McDermott, A temporal logic for reasoning about processes and plans, Cognitive Science 6 (1982) 101–155.

[32] K. L. McMillan, Symbolic Model Checking. An approach to the state explosion problem, PhD thesis, Carnegie-Mellon University, Pittsburgh, PA, USA, 1992.

[33] B. Nebel, H.-J. Bürckert, Reasoning about temporal relations: a maximal tractable subclass of Allen's interval algebra, Journal of the ACM, 42 (1) (1995) 43–66.

[34] O. Ore, Theory of Graphs, American Mathematical Society, Providence, RI, USA, 1962.

[35] J. Renz, Maximal tractable fragments of the region connection calculus: A complete analysis, in: Proceedings 16th International Joint Conference on Artificial Intelligence (IJCAI-99), Stockholm, Sweden, 1999, pp. 448–454.

[36] J. Renz, A spatial odyssey of the interval algebra: 1. directed intervals, in: Proceedings 17th International Joint Conference on Artificial Intelligence (IJCAI-2001), Seattle, WA, USA, 2001, pp. 51–56.

[37] J. Renz, B. Nebel, On the complexity of qualitative spatial reasoning: a maximal tractable fragment of the region connection calculus, Artificial Intelligence 108 (1–2) (1999) 69–123.

[38] R. Rodriguez, F. Anger, Using constraint propagation to reason about unsynchronized clocks, Constraints 2 (1998) 191–202.

[39] M. B. Vilain, H. A. Kautz, P. G. van Beek, Constraint propagation algorithms for temporal reasoning: a revised report, in: D. S. Weld, J. de Kleer, (Eds.), Readings in Qualitative Reasoning about Physical Systems, Morgan Kaufmann, San Mateo, CA, 1989, pp. 373–381.

[40] G. Winskell. An introduction to event structures. in: J. de Bakker, W.-P. de Roever, G. Rozenberg, (Eds.), Linear Time, Branching Time and Partial Order in Logics and Models for Concurrency, Springer-Verlag, 1989, pp. 123–172.

[41] M. Yannakakis, The complexity of the partial order dimension problem, SIAM Journal on Algebraic Discrete Methods, 3 (3) (1982) 351–358.