Algorithms and complexity for reasoning about time

Martin Charles Golumbic
IBM Israel Scientific Center
Technion City, Haifa, Israel; and
Bar-Ilan University, Ramat Gan, Israel
golumbic@israearn.bitnet

Ron Shamir
Computer Science Dept.
Tel Aviv University
Tel-Aviv 69978, Israel
shamir@math.tau.ac.il

Abstract

Interval consistency problems deal with events, each of which is assumed to be an interval on the real line or on any other linearly ordered set. This paper deals with problems in reasoning about such intervals when the precise topological relationships between them is unknown or only partially specified. This work unifies notions of interval algebras for temporal reasoning in artificial intelligence with those of interval orders and interval graphs in combinatorics, obtaining new algorithmic and complexity results of interest to both disciplines.

Several versions of the *satisfiability, minimum labeling* and *all consistent solutions* problems for temporal (interval) data are investigated. The satisfiability question is shown to be NP-complete even when restricting the possible interval relationships to subsets of the relations intersection and precedence only. On the other hand, we give efficient algorithm for several other restrictions of the problem. Many of these problems are also important in molecular biology, archaeology, and resolving mutual-exclusion constraints in circuit design.

1 Introduction

Reasoning about time is essential for applications in artificial intelligence and in many other disciplines. Given certain explicit relationships between a set of events, we would like to have the ability to infer additional relationships which are implicit in those given. For example, the transitivity of "before" and "contains" may allow us to infer information regarding the sequence of events. Such inferences are essential in story understanding, planning and causal reasoning. There are a great number of practical problems in which one is interested in constructing a time line where each particular event or phenomenon corresponds to an interval representing its duration. These include seriation in archeology [23, 24], behavioral psychology [9], temporal reasoning [1], scheduling [30], circuit design [38, p. 184] and combinatorics [31]. Indeed, it was the intersection data of time intervals that lead Hajös [21] to define and ask for a characterization of interval graphs, and which provides the clues for solving the "Berge mystery story" [16, p. 20]. Other applications arise in non-temporal context: For example, in molecular biology, arrangement of DNA segments along a linear DNA chain involves similar problems [6].

In this paper, we relate the two notions of interval algebra from the temporal reasoning community and interval graphs from the combinatorics community, obtaining new algorithmic complexity results of interest to both disciplines. Allen [1] defined a fundamental model for temporal reasoning where the relative position of two time intervals is expressed by the relations (less than, equal or greater than) of their four endpoints, generating thirteen primitive relations (see Table 1). We call this *13-valued interval algebra* $\mathcal{A}_{13}$. Our approach has been to simplify Allen's model in order to study its complexity using graph theoretic techniques. The first of the two lines of specialization which we study in this paper is macro relations. Macro relations refers to partitioning the 13 primitive relations into more coarse relations by regarding a subset of primitive relations as a new basic relation. We let

$$\cap = \{m, m^{-1}, o, o^{-1}, s, s^{-1}, f, f^{-1}, d, d^{-1}, \equiv\}$$

$$\alpha = \{m, o\}, \quad \alpha^{-1} = \{m^{-1}, o^{-1}\}$$

$$\subset = \{s, f, d\}, \quad \subset^{-1} = \{s^{-1}, f^{-1}, d^{-1}\}$$

From these we define the 3-valued and 7-valued interval algebras $\mathcal{A}_3$ and $\mathcal{A}_7$ whose elements are called its *atomic relations*, respectively,

$$\mathcal{A}_3 : \{\prec, \succ, \cap\} \qquad \mathcal{A}_7 : \{\prec, \succ, \alpha, \alpha^{-1}, \subset, \subset^{-1}, \equiv\}$$

For certain applications it is convenient to assume that all interval endpoints are distinct. This simplification

RELATION	NOTATION	INTERPRETATION
x before y	$\prec$	
y after x	$\succ$	
x meets y	m	
y met-by x	m^{-1}	
x overlaps y	o	
y overlapped-by x	o^{-1}	
x starts y	s	
y started-by x	s^{-1}	
x during y	d	
y includes x	d^{-1}	
x finishes y	f	
y finished-by x	f^{-1}	
x equals y	$\equiv$	

Table 1: The 13-valued interval algebra A_{13}. Thin line: x interval. Thick line: y interval.

generates the 6-valued algebra

$$\mathcal{A}_6 \ : \ \{\prec, \succ, o, o^{-1}, d, d^{-1}\}$$

The choice of which of these algebras $\mathcal{A}_i$ to use depends on the nature of the application, data, constraints and on the type of complexity result being proved.

We use the term "algebra" here since the set of atomic relations in each $\mathcal{A}_i$ forms a Boolean algebra, and in certain cases, together with an additional composition operation, forms a *relation algebra*, in the sense defined by Tarski [34], see [27].

In Section 2, we present the background on constraint satisfiability problems as they relate to the study of interval algebras. General results on the relative complexity of the problems of interval satisfiability, minimum labeling and finding all consistent solutions are presented in Section 3. Several new NP-completeness results are given in Section 4, specifically the interval satisfiability problem for the 3-valued interval algebra and the interval graph sandwich problem. Section 5 deals with polynomial time solutions to restricted domain problems. Our conclusions are given in Section 6.

2 Temporal reasoning as a constraint satisfiability problem

Temporal reasoning problems can be defined in the context of constraint satisfiability, where the variables will correspond to pairs of temporal events (i.e., intervals) and will take on values which represent the qualitative relationship between them (i.e., intersect, overlap, contain, less than, etc.) For each pair of events x and y, let $D(x,y)$ be a set of atomic relations in the algebra A_i. The semantics here is that we do not know precisely the relationship between x and y, but it must be one of those in the set $D(x,y)$. (In the language of constraint satisfiability, there is a variable $v(x,y)$ representing the relation between x and y, and its value must be taken from those in the set $D(x,y)$ corresponding to its *domain*.) For example, we read $D(x,y) = \{\prec, \subset\}$ as x is either before or contained in y.

The *interval satisfiability problem* (ISAT), called consistent singleton labeling in [35], is determining the existence of (and finding) one interval representation that is consistent with the input data $D(x,y)$. The *minimal labeling problem* (MLP) is to determine the minimal sets $D'(x,y) \subseteq D(x,y)$ such that every remaining atomic relation participates in some solution.

Example. $x\{\prec, m, o\}y$, $y\{\prec, \equiv, \succ\}z$, $z\{f, s\}x$.
Here xoy, $y{\succ}z$, zsx, and $x{\prec}y$, $y{\succ}z$, zfx are both consistent with the input, as shown in Figure 1. On the other hand, $y{\prec}z$ and $z \equiv y$ are impossible. The minimum labeling for this problem is $x\{\prec, m, o\}y$, $y{\succ}z$, $z\{f, s\}x$.

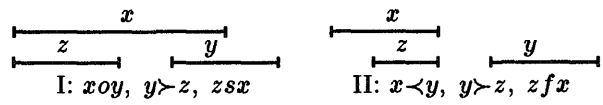

Figure 1: Two interval realizations for the above example

The *all consistent solutions problem* (ACSP), which we define here, is that of determining a polynomial representation structure Σ requiring $O(p(n))$ space and from which k distinct combinations of atomic relations consistent with the input can be produced in $O(q(n,k))$ time, where n is the number of variables, p and q are polynomial functions, and k is any number less than or equal to the number of solutions. The structure Σ thus represents *all* possible combinations of atomic relations consistent with the given data. Our contention is that the all consistent solutions problem is a more faithful closure problem for interval algebras than the minimal labeling problem, since not all tuples of the cross product of a minimal labeling are solutions. Consider the following simple example.

Example. $a\{\prec, \succ\}b$, $b\{\prec, \succ\}c$, $a\{\prec, \succ\}c$.

This is a minimal labeling, and yet only 6 of the possible 8 instantiations have interval representations.

The closely related *endpoint sequence problem* (ESP), is that of enumerating all the distinct interval realizations which are consistent with the given data. For $\mathcal{A}_{13}$ and $\mathcal{A}_6$, ACSP and ESP are equivalent, but in $\mathcal{A}_7$ and $\mathcal{A}_3$ there may be several (or many) distinct endpoint sequences which realize the same combination of atomic relations.

An application in archaeology: The *seriation* problem in archaeology attempts to place a set of artifact types in their proper chronological order. This problem was formulated by Flinders Petrie, a well-known archaeologist at the turn of the century, while studying 800 types of pottery found in 900 Egyptian graves. To each artifact type there corresponds a time interval (unknown to us) during which it was in use. To each grave there is a *point* in time (also unknown) when its contents were interred. Each grave provides the intersection data for the intervals corresponding to its contents.

3 The relative complexity of ISAT, MLP and ACSP

In this section we show that the polynomiality of the ISAT problem implies the polynomiality of MLP and ACSP. These results are valid in the general context of constraint satisfaction problems, if the maximum number of labels in a domain is bounded by a constant. We present here the results for the algebras $\mathcal{A}_i, i = 3, 6, 7, 13$, and they apply also to any restricted domain in these algebras.

Proposition 3.1 *The minimum labeling problem and the interval satisfiability decision problem are polynomially equivalent for each of the algebras $\mathcal{A}_i, i = 3, 6, 7, 13$.*

Proof. Clearly a solution to the MLP gives an answer to the ISAT decision problem. For the converse, one can use an oracle for ISAT to solve MLP as follows: Replace one relation set by one of the atomic relations it contains, and keep the rest of the problem input unchanged. ISAT is satisfiable for the resulting problem if and only if that atomic relation is part of a minimum labeling of the original problem. Hence MLP can be solved by a number of calls to ISAT oracle which equals the number of atomic relations in the input. ■

Proposition 3.2 *In any of the algebras $\mathcal{A}_i, i = 3, 6, 7, 13$, if the interval satisfiability problem is polynomial, then there exists a polynomial representation structure for the corresponding all consistent solutions problem.*

Proof. (Sketch) By Proposition 3.1, the polynomiality of ISAT implies the polynomiality of MLP. The algorithm for solving ACSP is an exhaustive depth-first search on the solution space defined by the cross product of the relation sets of the MLP solution. In each level of the search one more relation set is replaced by a singleton it contains, and the modified problem is checked for consistency using the ISAT oracle. This allows pruning partially restricted solutions which are already inconsistent at the root of their subtree without traversing it, and is the reason for the polynomiality of the algorithm. ■

4 Intractability of the complete algebras

Allen [1] originally provided a heuristic approach for solving the MLP in $\mathcal{A}_{13}$. That algorithm is polynomial but does not always provide a minimal solution, and may give a false positive answer to ISAT. Vilain and Kautz [37] have shown that MLP is in fact NP-complete for $\mathcal{A}_{13}$. Their proof relies on relations in which endpoints are equal, such as in the *meets* relation. We obtain a stronger result using macro relations to reduce the number of atomic relations from thirteen to three. Our first main result is to show that even in $\mathcal{A}_3$, the interval satisfiability problem is NP-complete. Consequently, all four problems ISAT, MLP, ACSP and ESP are intractable for all four interval algebras $\mathcal{A}_i$, $i = 3, 6, 7, 13$.

To prove that ISAT is NP-complete for $\mathcal{A}_3$, we introduce a new combinatorial problem, called the *interval graph sandwich problem*, which we prove NP-complete and show to be a special case of ISAT. An undirected graph $G = (V, E)$ is called an *interval graph* if its vertices can be represented by intervals on the real line such that two vertices are adjacent if and only if their intervals intersect (see [12, 13, 15, 16, 17].) The interval graph sandwich (IGS) problem is the following:

> **Interval Graph Sandwich problem:**
> INPUT: Two disjoint edge-sets, E^1 and E^2 on the same vertex set V.
> QUESTION: Is there a graph $G = (V, E)$ satisfying $E^1 \subseteq E \subseteq E^1 \cup E^2$ which is an interval graph?

When $E^1 = \emptyset$ or $E^1 \cup E^2$ is the complete graph on V, the answer is trivially yes. When $E^2 = \emptyset$, the problem is polynomial by the algorithm of Booth and Leuker [7]. The following new result shows that in the general case of the problem is NP-Complete.

Theorem 4.1 *The interval graph sandwich problem is NP-complete.*

The proof (omitted in this abstract) relies on a reduction from the Not-All-Equal 3-Satisfiability problem (Schaefer [32]), and the observation by Lekkerkerker and Boland [28] that an interval graph cannot contain an asteriodal triplet of vertices. The detailed proof is given in [19]. (se also [20].

Theorem 4.2 *ISAT is NP-complete for $\mathcal{A}_3$.*

Proof. Define $F = \{V \times V\} - \{E^1 \cup E^2\}$. For a given instance of the IGS problem, construct an instance of ISAT on $\mathcal{A}_3$ as follows: For each edge $(x, y) \in E^1$, E^2 or F, let $D(x, y) = \{\cap\}$, $\{\prec, \cap, \succ\}$ or $\{\prec, \succ\}$, respectively. It is clear that this ISAT problem has a solution if and only if the IGS has one. ■

Corollary 4.3 *ISAT, MLP, ACSP and ESP are NP-hard for $\mathcal{A}_3$, $\mathcal{A}_6$, $\mathcal{A}_7$ and $\mathcal{A}_{13}$.*

Proof. This follows from the observation that the algebra $\mathcal{A}_3$ is contained in $\mathcal{A}_i$ and that for any $i = 3, 6, 7, 13$, ISAT has a solution if and only if MLP has a non-empty solution if and only if ACSP has a non-empty solution if and only if ESP has a non-empty solution. ■

An application in molecular biology: In physical mapping of DNA, information on intersection or non-intersection of pairs of segments originating from a certain DNA chain is known from experiments. The goal is to find out how the segments can be arranged as intervals along a line (the DNA chain), so that their pairwise intersections in that arrangement match the experimental data. In the graph presentation, vertices correspond to segments and two vertices are connected by an E^1-edge (resp., F-edge) if their segments are known to intersect (resp., not to intersect). E^2-edges correspond the the case where the experimental information on the intersections is inconclusive, or simply unavailable. The decision problem is thus equivalent to the IGS problem.

5 Restricted domain problems

Because of the intractability of the general versions of ISAT, MLP ACSP and ESP, attention has been focused on the work of several authors who have studied polynomial time heuristic algorithms for MLP on $\mathcal{A}_{13}$ [1, 35, 36]. Solutions to several restricted cases of the interval satisfiability problem have been known for a long time. These will be extended by the new results presented in this section.

By suitably restricting the input domain of an NP-complete problem, one can often obtain a special class which admits a polynomial time algorithm. In the general case for an interval algebra $\mathcal{A}_i$, each relation set $D(x, y)$ may take any of $2^i - 1$ possible values (the empty subset of relations is not allowed). In this section, we restrict this by designating Δ to denote a particular family of relation sets in $\mathcal{A}_i$ and requiring that each set $D(x, y)$ be a member of Δ. To simplify notation we shall represent each relation set in $\mathcal{A}_3$ by a concatenation of its atomic relations, omitting braces. Hence $\prec\cap = \{\prec, \cap\}$, $\prec = \{\prec\}$ etc.. The seven possible relation sets in $\mathcal{A}_3$ in this notation are:

$$\prec, \succ, \cap, \prec\cap, \cap\succ, \diamond\!\!\!\!-, \prec\cap\succ.$$

ISAT(Δ) will denote the ISAT problem where all the relation sets are restricted to the set Δ. The proof of Theorem 4.2 shows that even when all relation sets are restricted to be from $\Delta_0 = \{\cap, \diamond\!\!\!\!-, \prec\cap\succ\}$ (meaning *intersect, disjoint* or *don't care*), ISAT(Δ_0) remains NP-complete.

A number of well-known recognition problems in graph theory and partially ordered sets may be viewed as restricted interval satisfiability problems. Five of these, all of which have polynomial time solutions, are given in Table 2 along with their appropriate Δ, see [12, 16, 31].

Class	Restricted Domain	Reference
Interval orders	$\{\prec, \succ, \cap\}$	[11]
Interval graphs	$\{\diamond\!\!\!\!-, \cap\}$	[15, 13, 7, 26]
Circle (or overlap) graphs	$\{\{\alpha, \alpha^{-1}, \equiv\}, \{\prec, \succ, \subset, \subset^{-1}\}\}$	[14, 8]
Interval containment graphs	$\{\{\subset, \subset^{-1}\}, \{\prec, \succ, \alpha, \alpha^{-1}, \equiv\}\}$	[18]
Posets of dimension 2	$\{\{\subset\}, \{\subset^{-1}\}, \{\prec, \succ, \alpha, \alpha^{-1}, \equiv\}\}$	[10, 2, 18]

Table 2: Polynomial interval satisfiability problems in graph theory.

Previous to the work of Belfer and Golumbic [3, 5, 4], we do not know of any study which has investigated the ESP. Their results demonstrate polynomial time solutions for the ESP in $\mathcal{A}_3$ restricted to (i) $\Delta = \{\prec, \succ, \cap\}$ (interval orders) using the so called Π structure and its associated construction algorithms, and (ii) $\Delta = \{\diamond\!\!\!\!-, \cap\}$ (interval graphs) using the endpoint-tree structure and its construction algorithms.

We describe here the results of a systematic study on the complexity of the restricted domains in $\mathcal{A}_3$. Since we can assume that two converse relation sets ($\prec$ and $\succ$, or $\prec\cap$ and $\cap\succ$) always appear together in a restricted domain, there are 31 possible restrictions. We classify 27 out of them as either polynomial or NP-complete, leaving open a conjecture that would settle the remaining four. For certain restricted domains special polynomial algorithms are devised.

5.1 Algorithms for the domain $\mathcal{A}_3 - \diamond\!\!\!\!-$.

In this section we deal with problems in the domain:

$$\Delta_1 = \{\prec, \succ, \cap, \prec\cap, \cap\succ, \prec\cap\succ\},$$

that is, $\mathcal{A}_3 - \diamond\!\!\!\!-$. We shall give efficient algorithms for ISAT, MLP and ACSP on this domain, and they will apply immediately to any subdomain of Δ_1. Hence by excluding just a *single* relation set from $\mathcal{A}_3$ the problems become tractable.

We first prove that ISAT(Δ_1) is polynomial. Construct a graph with vertices corresponding to the endpoints of event intervals and arcs representing the relative order of endpoints. The key observation is that every relation in Δ_1 is equivalent to a certain *order requirement* between a pair (or two pairs) of such endpoints. This observation is stated below:

Lemma 5.1 *Let i and j be the event intervals $[l_i, r_i]$ and $[l_j, r_j]$, respectively. In each of the following cases, the intervals satisfy the set of relations if and only if their endpoints satisfy the corresponding inequalities:*

$$i \prec j \;\Leftrightarrow\; r_i < l_j \qquad i \succ j \;\Leftrightarrow\; r_j < l_i \quad (1)$$
$$i \prec\!\cap j \;\Leftrightarrow\; l_i \le r_j \qquad i \cap\!\succ j \;\Leftrightarrow\; l_j \le r_i \quad (2)$$
$$i \cap j \;\Leftrightarrow\; l_i \le r_j \text{ and } l_j \le r_i \quad (3)$$

If $i \prec\!\cap\!\succ j$, no constraint is imposed.

The graph is now constructed as follows: For an instance J of ISAT(Δ_1) with n events, form a directed graph $G(J) = G(V, E)$ with vertex set $V = \{r_1, \ldots r_n, l_1, \ldots l_n\}$. The arc set E consists of two disjoint subsets, E_0 and E_1. The former will represent weak orders and the latter strict orders between pairs of endpoints. The arcs are defined as follows:

$$(l_i, r_i) \in E_0 \qquad i = 1, \ldots, n \quad (4)$$
$$(r_i, l_j) \in E_1 \qquad \forall i, j \; s.t. \; i \prec j \quad (5)$$
$$(l_i, r_j) \in E_0 \qquad \forall i, j \; s.t. \; i \prec\!\cap j \quad (6)$$
$$(l_i, r_j) \in E_0 \text{ and } (l_j, r_i) \in E_0 \qquad \forall i, j \; s.t. \; i \cap j \quad (7)$$

For pairs i, j with the relation $i \prec\!\cap\!\succ j$, no arc is introduced. Define now $E = E_0 \cup E_1$. We call the arcs in E_0 (resp., E_1) the *weak arcs* (resp., *strict arcs*). Note that the graph G is bipartite. Denote the two parts of vertex set by $R = \{r_1, \ldots, r_n\}$ and $L = \{l_1, \ldots, l_n\}$, and call arc an *RL-arcs* (resp., *LR-arc*) if it is directed from R to L (resp., from L to R). In the graph G all the RL-arcs are strict and all the LR-arcs are weak. Hence we need not record explicitly the type of each arc since it is implied by its direction. Since G is bipartite, a cycle must contain vertices both from R and from L. In particular, it must contain an RL-arc, so we obtain the following.

Lemma 5.2 *Every cycle in $G(J)$ contains a strict arc.* ■

Lemma 5.3 *Suppose $G(J) = (V, E)$ is acyclic. Then a linear order on V is consistent with the partial order $G(J)$ if and only if it is a realization of J.*

Proof. Take any linear order P which extends the partial order G. P is an ordering of all the endpoints, which by Lemma 5.1 satisfies all the relations in the input, so P gives a realization for J. On the other hand, every realization of J gives a linear order of the endpoints, in which each of the input relations must be satisfied, by Lemma 5.1. Hence the linear order must be consistent with the partial order $G(J)$. ■

An algorithm for solving ISAT(Δ_1) constructs $G(J)$ according to rules (4)-(7), and checks if it is acyclic. J is determined to be satisfiable if and only if $G(J)$ is acyclic.

Theorem 5.4 *The above algorithm correctly recognizes if an instance of ISAT(Δ_1) is satisfiable in linear time.*

Proof. *Validity:* By Lemma 5.1, each arc reflects the order relation of a pair of interval endpoints as prescribed by the input relations. If G contains a cycle, then by Lemma 5.2 that cycle contains a strict arc. Hence that cycle must satisfy $r_i < l_j \le \ldots \le r_i$, which implies that the input relations cannot be satisfied. In case G is acyclic, it represents a partial order on the vertices. By Lemma 5.3, it has a realization and hence J is satisfiable.

Complexity: Constructing $G(J)$ requires $O(m)$ steps, where m is the number of input relation sets, since the effort is constant per relation. Checking if G is acyclic can be done, for example, by depth first search, in time linear in $|E|$, the number of arcs [33]. Since $m \le n(n-1)/2$, and $|E| \le 2m + n$, the algorithm is linear. ■

Remark. Since all the constraints defining an instance of ISAT(Δ_1) are linear inequalities, the satisfiability problem can be reformulated as a feasibility question for a system of linear inequalities. This can be solved by linear programming algorithms, and in fact, by specialized algorithms using the fact only two variables appear in each inequality [29, 22]. While this is less efficient than the method described above, it allows the natural introduction of additional linear constraints, outside the scope of $\mathcal{A}_3$ or even $\mathcal{A}_{13}$, like lengths of intervals, fixing endpoints to specific time values, etc.

Our next result solves the Minimum Labeling Problem efficiently for the domain Δ_1. Once $G(J)$ has been constructed and shown to be acyclic, the MLP can be solved by forming the transitive closure of $G(J)$, deducing from it additional (weak and strict) orders of endpoints, and then using the equivalence established in Lemma 5.1 between these orders and interval relations to create the minimum labeling. The total complexity of the procedure is $O(mn)$ steps. A complete description appears in [19].

We finish this section by sketching a simple procedure to solve the ESP for Δ_1. The transitive closure of the graph $G(J)$ generated in the previous paragraph corresponds to a partially ordered set. The ESP thus reduces to constructing all the linear orders consistent with a partial order. This can be done by placing a minimal element in all possible positions with respect to a previously ordered subset of elements and repeating recursively. Using this technique we can show that all the realizations consistent with an instance on Δ_1 can be computed in $O(n)$ steps per each endpoint sequence produced. The distinction between strict and weak inequalities (following from the original distinction between strict and weak arcs) can also be maintained in such procedure.

5.2 The domain $\{\prec, \succ, \cap, \prec\!\cap\!\succ\}$.

Graph theoretic techniques provide a proof of the next result. An undirected graph is *chordal* if for every cycle of length greater than or equal to 4 there is an edge (chord) connecting two vertices which are not consecutive in the cycle. The complement $\overline{G}$ of G is the undirected graph whose edges are the non-edges of G; a graph is *transitively orientable* (TRO) if each undirected edge can be assigned a direction so that the resulting orientation satisfies the usual transitivity property. A classical characterization due to Gilmore and Hoffman [15] is that G is an interval graph if and only if G is a chordal graph and its complement $\overline{G}$ is transitively orientable.

Theorem 5.5 *ISAT is solvable in $O(n^3)$ time for $\Delta_2 = \{\prec, \succ, \cap, \diamond\!\!\!-\}$.*

Proof. Without loss of generality, we may assume that for each pair of elements x and y, the relation sets $D(x,y)$ and $D(y,x)$ given as input are already consistent, i.e., for each atomic relation R, $R \in D(x,y) \Leftrightarrow R^{-1} \in D(y,x)$. Otherwise, it is a simple matter to restrict the relation sets further so that they satisfy these properties or are shown to be unsatisfiable. Thus, for each pair of elements x and y, exactly one of the following holds:

(i) $x \cap y$ (ii) $x \diamond\!\!\!- y$ (iii) $x \prec y$ (iv) $x \succ y$.

We construct two complementary graphs G and H as follows. The graph $G = (V, E)$ has undirected edges where

$$\{x, y\} \in E \Leftrightarrow x \cap y.$$

The graph $H = (V, E')$ has both directed and undirected edges where

$$\{x, y\} \in E' \Leftrightarrow x \diamond\!\!\!- y$$
$$(x, y) \in E' \Leftrightarrow x \prec y$$

(An undirected edge between x and y is denoted by $\{x, y\}$. A directed edge from x to y is denoted by (x, y).) It is an easy consequence of the Gilmore-Hoffman Theorem that ISAT has a solution if and only if G is chordal and H has a transitive orientation.

Testing whether G is chordal can be done in $O(|V| + |E|)$ time [7], and obtaining a transitive orientation for H can be achieved in $O(|V||E|)$ time by a variant of the TRO algorithm [16, p. 124] for undirected graphs. An alternative polynomial solution to this problem has been given in [25]. ■

5.3 The domain $\{\prec, \succ, \prec\cap\succ, \diamond\!\!\!-\}$.

Theorem 5.6 *ISAT is polynomial for $\Delta_3 = \{\prec, \succ, \prec\cap\succ, \diamond\!\!\!-\}$.*

Proof. Form a directed graph $G(V, E)$ with vertices corresponding to events, and $(u, v) \in E$ if $u \prec v$. If G contains a cycle, then the instance is clearly not satisfiable. If G is acyclic, then one can create an interval realization of G in which (1) all intervals are disjoint, and (2) $(u, v) \in E$ if $u \prec v$. This can be done by taking any linear extension of G and ordering the intervals so that they are disjoint and ordered according to that order. In the resulting realization, (1) and (2) are satisfied. (2) guarantees that all $\prec$ and $\succ$ relations in the input are satisfied, and (1) guarantees that all other relations ($\diamond\!\!\!-$ and $\prec\cap\succ$) are satisfied. ■

5.4 The domain $\{\prec\cap, \cap\succ, \diamond\!\!\!-, \prec\cap\succ\}$.

The final theorem obtained here uses a reduction from the interval sandwich problem, to show the following intractability result.

Theorem 5.7 *ISAT is NP-complete for the restricted domain $\Delta_4 = \{\prec\cap, \cap\succ, \diamond\!\!\!-, \prec\cap\succ\}$.*

6 Conclusion

In this paper we have dealt with the consistency of assertions about the relations between intervals. We have investigated three basic problems in temporal reasoning: determining satisfiability (ISAT), maximum strengthening of a satisfiable assertion (MLP), and producing all the consistent solutions via a polynomial representation structure (ACSP and ESP). ACSP and ESP were shown to be tractable whenever ISAT is. We have shown that even a major simplification of Allen's interval algebra - from thirteen relations to three only - leaves ISAT intractable. On the positive side, we have shown that in this simplified algebra $\mathcal{A}_3$ many restricted domain problems are efficiently solvable. Of the 31 possible restrictions, we have classified 27 as either polynomial or NP-complete. A summary of the possible restrictions is presented in Figure 2. We conjecture that ISAT is NP-complete on $\Delta_5 = \{\prec\cap, \cap\succ, \diamond\!\!\!-\}$. A proof to this conjecture will resolve the remaining four cases. The tools we have used have been mainly from graph theory and complexity theory. We have hoped to demonstrate that the interconnection between these disciplines and reasoning problems in AI can be quite rich, and its investigation benefit both fields.

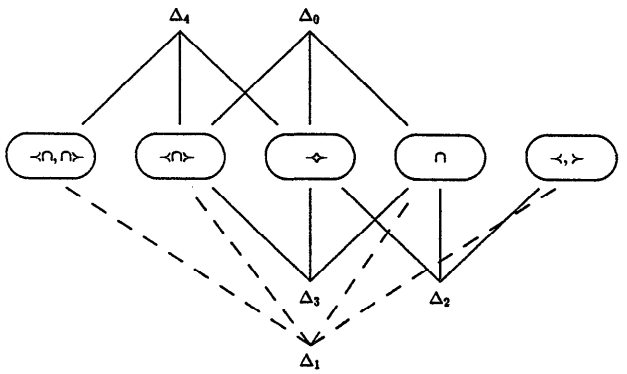

Figure 2: The complexity of ISAT on restricted domains of $\mathcal{A}_3$. Top sets are minimal NP-complete, bottom sets are maximal polynomial sets.

References

[1] J. F. Allen. Maintaining knowledge about temporal intervals. *Comm. ACM*, 26:832–843, 1983.

[2] K. R. Baker, P. C. Fishburn, and F. S. Roberts. Partial orders of dimension 2. *Networks*, 2:11–28, 1972.

[3] A. Belfer and M. C. Golumbic. Counting endpoint sequences for interval orders and interval graphs. *Discrete Math. (1992).*

[4] A. Belfer and M. C. Golumbic. The role of combinatorial structures in temporal reasoning. *Proc. AAAI Workshop on Constraint Satisfaction*, Boston, Mass., July 1990.

[5] A. Belfer and M. C. Golumbic. A combinatorial approach to temporal reasoning. In *Proc. Fifth Jerusalem Conf. on Information Technology*, pp. 774–780. IEEE Computer Society Press, 1990.

[6] S. Benzer. On the topology of the genetic fine structure. *Proc. Nat. Acad. Sci. USA*, 45:1607–1620, 1959.

[7] K. S. Booth and G. S. Lueker. Testing for the consecutive ones property, interval graphs, and planarity using PQ-tree algorithms. *J. Comput. Sys. Sci.*, 13:335–379, 1976.

[8] A. Bouchet. Reducing prime graphs and recognizing circle graphs. *Combinatorica*, 7:243–254, 1987.

[9] C. H. Coombs and J. E. K. Smith. On the detection of structures in attitudes and developmental processes. *Psych. Rev.*, 80:337–351, 1973.

[10] B. Dushnik and E. W. Miller. Partially ordered sets. *Amer. J. Math.*, 63:600–610, 1941.

[11] P. Fishburn. Intransitive indifference with unequal indifference intervals. *J. Math. Psych.*, 7:144–149, 1970.

[12] P. Fishburn. *Interval Orders and Interval Graphs.* Wiley, New York, 1985.

[13] D. R. Fulkerson and O. A. Gross. Incidence matrices and interval graphs. *Pacific J. Math.*, 15:835–855, 1965.

[14] C. P. Gabor, K. J. Supowit, and W-L Hsu. Recognizing circle graphs in polynomial time. *J. ACM*, 36:435–473, 1989.

[15] P. C. Gilmore and A. J. Hoffman. A characterization of comparability graphs and of interval graphs. *Canad. J. Math.*, 16:539–548, 1964.

[16] M. C. Golumbic. *Algorithmic Graph Theory and Perfect Graphs.* Academic Press, New York, 1980.

[17] M. C. Golumbic. Interval graphs and related topics. *Discrete Math.*, 55:113–121, 1985.

[18] M. C. Golumbic and E. R. Scheinerman. Containment graphs, posets and related classes of graphs. *Ann. N.Y. Acad. Sci.*, 555:192–204, 1989.

[19] M. C. Golumbic and R. Shamir. Complexity and algorithms for reasoning about time: A graph-theoretic approach. Technical Report 91-54, DIMACS Center, Rutgers University, NJ, 1991. Submitted for publication.

[20] M. C. Golumbic and R. Shamir. Complexity and algorithms for reasoning about time: A graph-theoretic approach. Proc. Israel Symp. on Theory of Computing and Systems, *Lecture Notes in Computer Science*, Springer-Verlag, 1992.

[21] G. Hajös. Über eine art von graphen. *Intern. Math. Nachr.*, 11, 1957. problem 65.

[22] D. S. Hochbaum and J. Naor. Simple and fast algorithms for linear and integer programs with two variables per inequality. Proc. 2nd Conf. on Integer Programming and Combinatorial Optimization Carnegie Mellon University Press, 1992, to appear.

[23] D. G. Kendall. Incidence matrices, interval graphs, and seriation in archaeology. *Pacific J. Math.*, 28:565–570, 1969.

[24] D. G. Kendall. Some problems and methods in statistical archaeology. *World Archaeol.*, 1:68–76, 1969.

[25] N. Korte and R. H. Möhring. Transitive orientation of graphs with side constraints. In H. Noltemeier, editor, *Proc. of the International workshop on Graphtheoretic concepts in computer Science (WG '85)*, pp. 143–160, Linz, 1985. Universitätsverlag Rudolf Trauner.

[26] N. Korte and R. H. Möhring. An incremental linear time algorithm for recognizing interval graphs. *SIAM J. Comput.*, 18:68–81, 1989.

[27] P. B. Ladkin and R. Maddux. The algebra of constraint satisfaction problems and temporal reasoning. Technical report, Kestrel Institute, Palo Alto, 1988.

[28] C. G. Lekkerkerker and J. Ch. Boland. Representation of a finite graph by a set of interval on the real line. *Fundam. Math.*, 51:45–64, 1962.

[29] N. Megiddo. Towards a genuinely polynomial algorithm for linear programming. *SIAM J. Comput.*, 12:347–353, 1983.

[30] C. Papadimitriou and M. Yannakakis. Scheduling interval ordered tasks. *SIAM J. Comput.*, 8:405–409, 1979.

[31] F. S. Roberts. *Discrete Mathematical Models, with Applications to Social Biological and Environmental Problems.* Prentice-Hall, Englewood Cliffs, New Jersey, 1976.

[32] T. J. Schaefer. The complexity of satisfiability problems. In *Proc. 10th Annual ACM Symp. on Theory of Computing*, pp. 216–226, 1978.

[33] R. E. Tarjan. Depth-first search and linear graph algorithms. *SIAM Journal on Computing*, 1:146–160, 1972.

[34] A. Tarski. On the calculus of relations. *Journal of Symbolic Logic*, 6:73–89, 1941.

[35] P. vanBeek. Approximation algorithms for temporal reasoning. In *Proc. Eleventh Int'l. Joint Conf. on Artificial Intelligence (IJCAI-89)*, pp. 1291–1296, August 1989.

[36] P. vanBeek. Reasoning about qualitative temporal information. In *Proc. Eighth Nat'l. Conf. on Artificial Intelligence (AAAI-90)*, pp. 728–734, August 1990.

[37] M. Vilain and H. Kautz. Constraint propagation algorithms for temporal reasoning. In *Proc. Fifth Nat'l. Conf. on Artificial Intelligence (AAAI-86)*, pp. 337–382, August 1986.

[38] S. A. Ward and R. H. Halstead. *Computation Structures.* MIT Press, 1990.