On Finding a Solution in Temporal Constraint Satisfaction Problems

Alfonso Gerevini
Dipartimento di Elettronica per l'Automazione
Universita degli studi di Brescia
Via Branze 38, 1-25123 Brescia, Italy
gereviniQing.unibs.it

Matteo Cristani
Istituto Policattedra
Universita degli studi di Verona
Strada le Grazie, 1-37134 Verona, Italy
cristani@romeo.sci.univr.it

Abstract

Computing a consistent interpretation of the variables involved in a set of temporal constraints is an important task for many areas of AI requiring temporal reasoning. We focus on the important classes of the qualitative relations in Nebel and Biirckert's ORD-Horn algebra, and of the metric constraints forming a STP, possibly augmented with inequations. For these tractable classes we present three new algorithms for solving the problem of finding a solution, and an efficient algorithm for determining the consistency of a STP augmented with inequations.

1 Introduction

Reasoning about temporal constraints is an important task in many areas of AI. Allen's Interval Algebra (IA) [1983] and Vilain and Kautz's Point Algebra (PA) [1986] are two fundamental models for qualitative temporal reasoning, while Dechter, Meiri and Pearl's TCSP [1991] is a prominent approach for metric temporal reasoning.

Given a set S of temporal constraints, two important related reasoning problems are determining the consistency of S, and finding a consistent scenario or solution for the variables involved in S. A consistent scenario is an ordering of the variables (either points or interval endpoints) in S, which is consistent with the constraints in S. A solution for S is an interpretation of the point-variables (interval-endpoint variables) in S which satisfies'the constraints in S.

Consistency checking and finding a solution (a consistent scenario), are NP-Hard problems for IA [Vilain and Kautz, 1986] and TCSP [Dechter *et al,* 1991], while they are polynomial for PA [van Beek, 1990] and for some important restrictions of IA and TCSP. These include the qualitative interval relations of the ORD-Hbrn class [Nebel and Burckert, 1995], and the metric constraints of a "simple temporal constraint satisfaction problem" (STP) [Dechter *et* al., 1991]. The ORD-Horn class forms a subalgebra of IA, which is the maximal tractable subclass of relations in IA containing all the thirteen basic relations. The constraints of a STP are inequalities of the form $p2 - p1 \leq d$, where p1,p2 are point-variables, and *d* is any value in a dense time domain.[1]

$\mathrm{STP}^{\neq}$ is another interesting tractable class, which subsumes PA and STP. $\mathrm{STP}^{\neq}$ is an extension of STP to include inequations, i.e. constraints of the form $p2 - p1 \neq d$ [Koubarakis, 1995; Gerevini and Cristani, 1995]. Koubarakis [1992] proposed a method for checking the consistency of a STP augmented with disjunctions of inequations, whose time complexity reduces to $O(n^3 + kn^2)$ when the input is a $\mathrm{STP}^{\neq}$, where *n* is the number of variables and *k* the number of inequations.

In this paper we are mainly concerned with the problem of finding a solution for these tractable classes. While Nebel and Burckert [1995] proved several interesting strong results about their ORD-Horn algebra, they left open the problem of efficiently finding a consistent scenario or solution.

Dechter *et al.* [1991] proposed a simple algorithm for finding a solution of a given STP which only contains *non strict* inequalities, leaving open the important case in which inequalities can be strict (e.g., $p2 - p1 < d$).

Concerning $\mathrm{STP}^{\neq}$, we are not aware of any specialized algorithm for the problem of finding a solution.

We propose three new algorithms for solving these problems. Also, we investigate the problem of determining the consistency of a given $\mathrm{STP}^{\neq}$, presenting an efficient algorithm which improves the complexity bound of Koubarakis' method. Specifically, we will present:

- a simple algorithm for finding a solution for a set of ORD-Horn relations. The algorithm requires $0(n^2)$ time, if the input set of relations is known to be path-consistent, and $0(n^3)$ time in the general case, where *n* is the number of interval-variables.
- An $0(n^3)$ time algorithm for finding a solution for a given STP including *strict* inequalities.
- An $0(n^3 + k)$ time algorithm for determining the consistency of a given $\mathrm{STP}^{\neq}$.[2]

[1] Equality constraints such as $p2 - p1 = d$ can be expressed as a pair of inequalities.

[2] Note that when the number of input inequations *(k)* is limited to those required to express the qualitative point relations of PA or of the interval relations in the "Pointizable

- An $O(n^3 + k)$ time algorithm for finding a solution for a given $STP^{\neq}$.

2 Finding a solution for ORD-Horn interval relations

In this section we provide an algorithm for finding a solution of a set of relations over the ORD-Horn interval algebra ($\mathcal{H}$). The time complexity of our algorithm is $O(n^2)$, if the input set of relations is known to be path-consistent, and $O(n^3)$ in the general case, where n is the number of interval variables.

The proofs of our claims are based on the following definitions and facts:

(1) Disjunctions of PA-relations of the form $a = b$, $a \leq b$, $a \neq b$ are called ORD clauses. ORD clauses containing at most one literal (PA-relation) of the form $a = b$ or $a \leq b$ and any number of literals of the form $a \neq b$ are called ORD-Horn clauses [Nebel and Burckert, 1995].

(2) The theory *ORD* that axiomatizes "=" as an equivalence relation and "$\leq$" as a partial ordering over the equivalence classes is a Horn theory [Nebel and Burckert, 1995].

(3) Let Θ be a path-consistent set over $\mathcal{H}$. Then $\pi(\Theta) \cup ORD_{\pi(\Theta)}$ does not allow the derivation of new unit clauses by positive unit resolution, where $\pi(\Theta)$ denotes the set of ORD-Horn clauses translating the relations in Θ, and $ORD_{\pi(\Theta)}$ denotes the axioms of ORD instantiated to all the point-variables mentioned in $\pi(\Theta)$ [Nebel and Bürckert, 1995].

(4) Let Θ be a set of relations over $\mathcal{H}$. Each clause of $\pi(\Theta)$ is either unary or binary [Nebel and Bürckert, 1995].

(5) A temporally labeled graph (*TL-graph*) is a graph with at least one vertex and a set of labeled edges, where each edge (v, l, w) connects a pair of distinct vertices v, w. The edges are either directed and labeled $\leq$ or $<$, or undirected and labeled $\neq$ [Gerevini and Schubert, 1995].

(6) A set S of the PA-relations can be represented by a TL-graph T such that S is consistent *iff* T does not contain any $<$-cycle, or any $\leq$-cycle that has two vertices connected by an edge with label $\neq$ [Gerevini and Schubert, 1995; van Beek, 1990].[3]

(7) A set S of PA-relations can be translated into a logically equivalent set of unary ORD-Horn clauses [Nebel and Bürckert, 1995], which we will call the *ORD-Horn translation* of S.

Before introducing our algorithm, we need to prove the following lemma:

subclass of IA" [Ladkin and Maddux, 1988], our algorithm takes $O(n^3)$ time, while Koubarakis's method requires $O(n^4)$ time. If k is larger than $O(n^2)$, then the improvement is even more significant.

[3] A =-relation such as $x = y$ is represented as the pair of edges $(x, \leq, y)$ and $(y, \leq, x)$.

Lemma 1 *Let Σ be the ORD-Horn translation of a consistent set S of PA-relations, and x and y two point-variables involved in S. If S entails $x = y$ then $x = y$ is derivable from $\Sigma \cup ORD_{(\Sigma)}$ by positive unit resolution.*

Proof. Suppose that S entails $x = y$, but $x = y$ cannot be derived from $\Sigma \cup ORD_{(\Sigma)}$ by positive unit resolution. Then, the set $\Sigma' = \Sigma \cup \{x \neq y\}$ is inconsistent, but from $\Sigma' \cup ORD_{(\Sigma')}$ we cannot derive the empty clause by positive unit resolution (because the only positive unit resolution possible against $x \neq y$ is with positive unit clause $x = y$). Since $\Sigma' \cup ORD_{(\Sigma')}$ is a set of propositional Horn clauses, this contradicts the fact that positive unit resolution is refutation-complete for propositional Horn theories [Henschen and Wos, 1974].

Therefore, if S entails $x = y$, then $x = y$ must be derivable from $\Sigma \cup ORD_{(\Sigma)}$ by positive unit resolution. □

Theorem 1 *There exists an $O(n^2)$ time algorithm for finding a consistent scenario of a path-consistent set Θ of relations over $\mathcal{H}$, where n is the number of interval variables involved in Θ.*

Proof (sketch). By Property (4) we have that $\pi(\Theta) = \Sigma_1 \cup \Sigma_2$, where Σ_1 is a set of unary Horn clauses and Σ_2 is a (possibly empty) set of binary Horn clauses, each of which has at least one $\neq$-disjunct ($\neq$-relation).[4]

Let T be the TL-graph representing Σ_1, and P the set of the positive unit clauses entailed by Σ_1. By properties (3–4) and Lemma 1, we have that if $x \neq y$ is a disjunct of a clause in Σ_2, then $(x = y) \notin P$.

Consider now choosing a $\neq$-disjunct for each of the clauses in Σ_2. Let I be the set of inequations selected, and T' the TL-graph obtained by extending T with the edges representing the relations in I. Since T is consistent, by Property (6) and construction of T', T' does not contain any $<$-cycle. Furthermore, by construction of Σ_1, Σ_2 and T, if $x \neq y$ is a disjunct in Σ_2 then T does not contain a $\leq$-cycle crossing x and y (otherwise we would have that $(x = y) \in P$). Hence, by Property (6) $\Sigma_1 \cup$ I is consistent.

It is easy to see that the endpoints of all the interval variables involved in Θ have a corresponding point-variable in $\Sigma_1 \cup$ I, and that if we find a solution for $\Sigma_1 \cup$ I, then we will also have a solution for Θ. In fact, if s is a solution for $\Sigma_1 \cup$ I, then, by construction of $\Sigma_1 \cup$ I, the interpretation of the interval endpoints defined by s do satisfy $\pi(\Theta)$ and Θ.

A consistent scenario s for a set of PA-relations involving m point-variables can be found in $O(m^2)$ time [van Beek, 1990]. From s we can derive a solution in $O(m)$ time (e.g., we assign an integer i to the variables in the first position of s, $i + 1$ to the variables in the second position, etc.). Since $\Sigma_1 \cup$ I is a set of PA-relations involving $2n$ variables (those corresponding to the endpoints of the n interval variables of Θ), we can find a solution for $\Sigma_1 \cup$ I (and for Θ) in $O(n^2)$ time. □

[4] $\pi(\Theta)$ can be computed in $O(|\Theta|)$ time using a table of 868 elements containing the translation of each ORD-Horn relation.

Let $\pi_1(\Theta)$ be the set of unary clauses in $\pi(\Theta)$, and $\pi_2(\Theta)$ the set of binary clauses in $\pi(\Theta)$. Given a set Ω of relations in $\mathcal{H}$, from the proof of Theorem 1, and the fact that the consistency of Ω can be decided in $O(n^3)$ time by enforcing path-consistency to it [Nebel and Bürckert, 1995], it is easy to see that the following algorithm computes a solution for Ω (if it exists) in $O(n^3)$ time:

Algorithm 1: ORD-HORN-SOLUTION
Input: a set Ω of relations in $\mathcal{H}$.
Output: a solution for Ω, if one exists; `nil` otherwise.

1. Determine the consistency of Ω by enforcing path-consistency to it. If Ω is not consistent, then return `nil`, otherwise let Θ be the path-consistent set of relations equivalent to Ω.
2. Run Beek's algorithm [1990] for determining a consistent scenario of a set of PA-relations on $\pi_1(\Theta) \cup D$, where D is a set of $\neq$-relations consisting of a $\neq$-disjunct for each clause in $\pi_2(\Theta)$. Let s be the scenario computed.[5]
3. Derive from s a solution for Θ (Ω) by assigning to each interval endpoint of Θ a number consistent with s. (This can be done in $O(n)$ time, as discussed in the proof of Theorem 1.)

Therefore, we have proved the following theorem:

Theorem 2 *Given a set Ω of relations in $\mathcal{H}$,* ORD-HORN-SOLUTION *computes a solution for Ω (if it exists) in $O(n^3)$ time, where n is the number of variables in Ω.*

Remark. If the input set of relations is known to be path-consistent, then step 1 can be omitted and the time complexity of ORD-HORN-SOLUTION reduces to $O(n^2)$.

3 Preliminaries on STP and STP$^{\neq}$

Most of the terminology introduced in this section is based on the concept of a Temporal Constraint Network given in [Dechter *et al.*, 1991].

A *Temporal Constraint Network* (*TN*) is a directed labeled graph where the vertices represent point variables over a *dense* time domain T (e.g., the real numbers), and the edges connect distinct vertices and are labeled by a finite set of convex intervals over T. An edge (v, l, w) from v to w with label $\{I_1, I_2, ..., I_n\}$ represents the binary constraint: $C_{vw} = w - v \in I_{vw}$, where $I_{vw} = \bigcup_{i=1}^{n} I_i$.

The constraints of a given STP can be represented by a *Simple Temporal Network* (*STN*), which is a TN where each edge is labeled by exactly one (convex) interval [Dechter *et al.*, 1991].

Given a set C of constraints represented by a TN (TN-constraints) involving the set of variables $x_1, x_2, .., x_n$, the set of values $i_1, i_2, .., i_n$ is a *solution* of C *iff* the assignment $x_1 = i_1, i_2 = i_2, .., x_n = i_n$ satisfies all the constraints in C. A TN is *consistent iff* the corresponding set of constraints has at least one solution. Two TNs are *equivalent* if they have the same set of solutions.

Given two TN-constraints C_{xy} and C'_{xy}, involving the same pair of variables, C_{xy} is *tighter* than C'_{xy} ($C_{xy} \subseteq C'_{xy}$) when $I_{xy} \subseteq I'_{xy}$ [Dechter *et al.*, 1991]. Given two TNs $\mathcal{S}$ and $\mathcal{T}$, $\mathcal{S}$ is tighter than $\mathcal{T}$ ($\mathcal{S} \subseteq \mathcal{T}$) if every constraint represented by $\mathcal{S}$ is tighter than the corresponding constraint represented by $\mathcal{T}$. If $\mathcal{S}$ is tighter than $\mathcal{T}$ then the solutions of $\mathcal{S}$ are also solutions of $\mathcal{T}$. The *minimal network* of a TN $\mathcal{T}$ is the tightest network equivalent to $\mathcal{T}$ [Dechter *et al.*, 1991; Montanari, 1974].

A TN T is *decomposable* [Dechter *et a*/., 1991; Montanari, 1974] if every locally consistent assignment to any subset of variables (vertices) of T can be extended to a solution.[6]

To prove our results we will also use the following definitions and properties from [Dechter *et a*/., 1991]:

(8) Given a STN *S* the *distance graph* of *S* is a directed labeled graph with the same vertices as S, and with an edge from *v* to *w,* labeled by the upper bound of the interval of the label on the edge from *v* to *w* in *S* and an edge from *w* to ?;, labeled by -1 times the lower bound of the interval of the label on the edge from *v* to *w* in *S.*

(9) A STN is consistent *iff* its distance graph does not contain "negative cycles", i.e., cycles where the sum of the labels on the edges is negative.

(10) Given a STN *S* the *d-graph* of S is a directed labeled graph that has the same vertices as S, and an edge for each pair of (not necessarily distinct) vertices in *S* labeled by the shortest path between those vertices in the distance graph. *S* is consistent *iff* its d-graph does not contain any negative edge from a vertex to itself (a circular edge).

Note that Property (9) assumes that the intervals of the labels of the STN are all closed, except the intervals where the lower bound is $-\infty$, or the upper bound is $+\infty$ (i.e., in the STP represented by the STN there are no *strict* inequalities of the form *y-x* <*d).* Such an assumption is made for simplicity in [Dechter *et a*/., 1991, p. 64]. When the intervals can be (semi)open (i.e., in the STP there are strict inequalities), in order to ensure the consistency of the STN (STP), the absence of negative cycles in the distance graph corresponding to the STP can still be a sufficient condition, provided that the notion of negative cycle is slightly extended to take into account the presence of the strict inequalities. This can easily be done by using a method similar to the method used by Kautz and Ladkin [1991].

The constraints of a STP$^{\neq}$ can be represented by a *STN augmented with inequations,* defined as follow:

[5] Note that since $\pi(\Theta) \cup D$ is known to be consistent, we could actually consider only the PA-relations of $\pi_1(\Theta)$. This is because any consistent scenario computed by applying van Beek's algorithm to $\pi_1(\Theta)$ is also a consistent scenario for $\pi_1(\Theta) \cup D$.

[6] An assignment of values to a set *S* of variables is locally consistent if it satisfies the constraints involving only variables in *S* [Dechter *et al.,* 1991].

Definition 1 *A* STN augmented with inequations *($STN^{\neq}$) is a directed labeled graph where the label on each edge is a pair $\langle T, E\rangle$, T is a convex interval that can be either closed, semi-open or open with the open bound equal to $+\infty$ or $-\infty$, and E is a finite (possibly empty) set of points of T called* excluded points *of T.*

The label $\langle T_{vw}, E_{vw}\rangle$ on an edge $(v, \langle T_{vw}, E_{vw}\rangle, w)$ from v to w of a $STN^{\neq}$ represents the constraint $w - v \in \{T_{vw} - E_{vw}\}$.

A $STN^{\neq}$ T *entails* $w - v \neq d$ ($w - v = d$) *iff* v and w are vertices of T, and there is no solution for the set of constraints represented by T such that $w - v = d$ ($w - v \neq d$) is satisfied.

Definition 2 *Given a $STN^{\neq}$ T, the STN obtained from T by substituting $[a, b]$ for any label $\langle [a, b], E\rangle$ of T is called the* relaxed network *of T, written as T^r.*

Definition 3 *Given a $STN^{\neq}$ T and two vertices v,w of T such that $I = [I^-, I^+]$ is the label of the edge from v to w in the minimal network of the relaxed network of T, an inequation $w - v \neq d$ of T is:*

- convex iff $d = I^-$ *or* $d = I^+$;
- non-convex iff $I^- < d < I^+$.

4 Finding a solution for STP

In this section we address the problem of finding a solution for a given STP. The consistency of a STP S can be determined in $O(n^3)$ time, where n is the number of variables in S [Dechter *et al.*, 1991]. If S is not consistent, then there are no solutions for it. If S is consistent, then we distinguish two cases: S does not contain strict inequalities, and S does contain strict inequalities.

In the first case, the following is a solution that can be computed in $O(n^3)$ time [Dechter *et al.*, 1991]:

$$x_0 = 0, x_1 = l_1, ..., x_n = l_n,$$

where x_i ($i = 1..n$) are the variables of the given STP, x_0 is a special additional variable indicating the absolute starting time and preceding all the other variables ($0 \leq x_i - x_0 < +\infty$), and l_i is the lower bound of the interval labeling the edge from x_0 to x_i in the minimal network of the given STP.[7]

If S is consistent and contains strict inequalities, then we prove that the following algorithm computes a solution in $O(n^3)$ time:

Algorithm 2: STP-SOLUTION
Input: a STP S
Output: a solution for S, if it exists; `nil`, otherwise

0. Check the consistency of S. If S is not consistent then return `nil` and stop.
1. Compute the minimal network M of the STP derived from the original one by relaxing each strict inequality of the form $y - x < d$ (d finite) to the *non*-strict inequality $y - x \leq d$.
2. For each interval I labeling an edge of M, if it has an *open* and finite left bound I^-, then replace it with the *closed* bound $I^- + \epsilon$, where ϵ is a positive quantity defined as:[8]
$$\epsilon = \frac{\delta}{n^2 + 1}$$
and δ is the *finite* length of the shortest interval labeling the edges of M, if this exists, or any finite number if all the intervals of M have $-\infty$ or $+\infty$ as one of their bounds. Let us call the resulting relaxed network M'.
3. Compute the minimal network M'' of M'.
4. Return $x_0 = 0, x_1 = l_1, ..., x_n = l_n$ as a solution for S (M), where l_i are the left (closed) bounds of the intervals labeling the edges from x_0 to x_i in M'' ($i = 1..n$).

Theorem 3 *STP-*CONSISTENCY *computes a solution for a given STP (STN) in $O(n^3)$ time (if it exists), where n is the number of point-variables (vertices).*

Proof. It is easy to see that the global time complexity of STP-SOLUTION is $O(n^3)$. Concerning its correctness we observe that:

(i) all the intervals of M' with a finite left bound are closed on the left (by construction of the network). Hence in M'' the left bound of the intervals on the edges from x_0 to x_i ($i = 1..n$) will also be closed.
(ii) All the solutions of M' are also solutions of M (because M' is tighter that M).
(iii) M' is consistent (provided that the original STP was consistent).

In order to prove (iii) suppose that step 2 adjusts k left-open intervals of M to derive M'. Consider now performing each of these perturbations incrementally (instead of all at once), according to the following scheme:

1'. $N := M$;
2'. adjust one of the finite left-open bounds I_N^- of N by increasing it to $I_N^- + \epsilon$;
3'. $M :=$ Minimal network of N;
4'. if M contains a finite open left bound then goto 1' else return M.

Note that after each iteration in this scheme the number of intervals in M with a finite open left bound monotonically decreases. So, a sequence of at most k "tuned" networks is computed at step 2', and at most k minimal networks are computed at step 3'. (The actual number of these networks can be less than k, because the computation of the minimal network of N can make closed some open left bounds.)

Let $N_1, ..., N_h$ be the sequence of networks computed at step 2' and $M_1, ..., M_h$ the sequence of minimal networks computed at step 3' ($1 \leq h \leq k \leq n^2$). We can prove by induction on h that the tuned network N_h is

[7] Note that in such a minimal network all the left bounds on the distance between x_0 and x_i are finite.

[8] The use of appropriate ϵ-quantities is a known technique in Linear Programming.

consistent. This is obvious for $h = 1$, since M is decomposable. Suppose now that N_{h-1} is consistent (and M_{h-1} is minimal), we show that N_h is consistent as well (M_h is minimal). We have that in M_{h-1} any finite left bound interval (either open or closed) can be increased at most to $I_M^- + (h-1)\epsilon$, relevant to its value (I_M^-) in the original network M (by construction of M_{h-1}). While the right interval bound of any interval of M_{h-1} maintains the same value as in the original M. Since by definition of ϵ we have that

$$h\epsilon = \frac{h\delta}{n^2+1} \leq \frac{n^2\delta}{n^2+1}$$

and by definition of δ $I_M^- + h\epsilon < I_M^+$, we can increase the (finite) left open bound of any interval in M_{h-1} of ϵ, deriving a new left bound which is still strictly less than the corresponding right bound in M_{h-1}. Hence, since M_{h-1} is minimal and decomposable, the tuned network N_h will be consistent.

To conclude, note that the network returned by the "incremental" algorithm *(Mh)* is the same as the network M" computed at step 3 of the "non-incremental" algorithm, and that the existence of ϵ is guaranteed by the fact that the temporal variables have dense domains. □

5 Checking consistency and finding a solution for STP$^{\neq}$

In this section we first prove that checking the consistency of a STN$^{\neq}$ can be accomplished in $O(n^3 + k)$ time and $O(n^2 + k)$ space, where n is the number of point-variables, and k the number of inequations. Then, we give an algorithm for finding a solution of a given STP$^{\neq}$, which has the same complexity as consistency checking.

The following proposition is a direct consequence of property (9) given in Section 3. It will be used in the proof of Lemma 2, which is the base for proving the correctness of our algorithm for checking the consistency.

Proposition 1 *A STN$^{\neq}$ T is consistent* iff *for each inequation $w - v \neq d$ it is possible to choose one of the two inequalities $w - v < d$ or $v - w < -d$ so that the distance graph obtained by extending the distance graph of T^r with the resulting inequalities does not contain negative cycles.*

Lemma 2 *A STN$^{\neq}$ T is consistent* iff *T^r does not have negative cycles in its distance graph, and it does not entail $w - v = d$ for any inequation $w - v \neq d$ in T.*

Proof. The "only if" direction is trivial. The "if" direction is proved by induction on the number k of inequations of T. We first consider the case in which for every inequation $w - v \neq d$, $d \geq 0$. When $k = 0$ the absence of negative cycles in the distance graph guarantees consistency. Suppose that the property holds for $k - 1$ inequations and consider a STN$^{\neq}$ $\mathcal{N}$ with $k - 1$ inequations, whose relaxed network $\mathcal{N}^r$ does not have any negative cycles in its distance graph, and does not entail the equation corresponding to any of the $k - 1$ inequations. Thus, by the induction hypothesis $\mathcal{N}$ is consistent, and

Algorithm 3: STN$^{\neq}$-CONSISTENCY
Input: a STN$^{\neq}$ T
Output: true if T is consistent, nil otherwise

1. Compute the d-graph D of the relaxed network of T;
2. if D contains negative circular edges then return nil;
3. for each inequation $w - v \neq d$ in T do
   if the label on the edge of D from *v* to *w* is *d* and
   the label on the edge from *w* to *v* is $-d$
   then return nil;
4. return true.

Figure 1: STN$^{\neq}$-CONSISTENCY

by Proposition 1 we can extend the distance graph of N^r with one inequality for each inequation of N, obtaining a graph *D* without negative cycles.

Consider adding the inequation $w - v \neq d$ to *N*. We can do that consistently if we can consistently add $w - v < d$ or $v - w < -d$ to D. Since when there is a path from *v* to *w* in *D* there is also a path from *w* to *v*, and since *D* does not contain negative cycles, only one of the following possibilities can hold:

(a) all the paths connecting *v* and *w* are positive paths;

(b) there are negative (and possibly positive) paths from *w* to *v* and only positive paths from v to *w;*

(c) there are negative (and possibly positive) paths from *v* to *w* and only positive paths from *w* to *v*.

In cases (a) and (b) we can add $w - v < d$ to D without creating a negative cycle, so that by Proposition 1 $\mathcal{N} \cup \{w - v \neq d\}$ is consistent. For the remaining case (c) suppose that both the addition of $w - v < d$ and of $v - w < -d$ create a negative cycle. Then there exists a negative path of length l (with $l < 0$) from *v* to *w* and a positive path of length l' (with $l' > 0$) from *w* to *v* such that $d + l < 0$ and $l' - d < 0$. From these inequalities we derive $l + l' < 0$. But this means that *D* contains a negative cycle, contrary to the construction of *D*. Hence, also for case (c) at least one of $w - v < d$ and $v - w < -d$ can be consistently added to *D*. It follows that $w - v \neq d$ can be consistently added to *N*,

The proof can be extended to include inequations $w - v \neq d$ ($d < 0$) excluding negative points by an analogous argument, using induction on the number of such inequations. In cases (a) and (c) we can consistently add $v - w < -d$, while we deal with case (b) in the same way as we did for case (c) when $d > 0$. □

The previous lemma guarantees that consistency checking of a STN$^{\neq}$ (STP$^{\neq}$) can be accomplished by using the algorithm given in Figure 1.

Theorem 4 *STN$^{\neq}$* -CONSISTENCY *correctly checks the consistency of a STN$^{\neq}$ T in $O(n^3+k)$ time and $O(n^2+k)$ space, where n is the number of the vertices of T and k is the number of inequations.*

Proof (sketch). Property (10) of STNs (see Section 3) together with Lemma 2 ensure the correctness of STN$^{\neq}$ CONSISTENCY. Regarding the complexity, it is sufficient to observe that: the relaxed network of T can be computed in time linear over the number of the edges of T; step 1 can be accomplished in $0(n^3)$ time and $0(n^2)$ space [Cormen *et al.,* 1990; Dechter *et a*l., 1991], step 2 in $0(n)$ time, and step 3 in $O(k)$ time and space. D

We now show that the problem of finding a solution for a given STP$^{\neq}$ can be solved by running an algorithm which has the same complexity as STN$^{\neq}$-CONSISTENCY.

Theorem 5 *A solution for a given STN$^{\neq}$ (STP$^{\neq}$) can be found in* $0(n^3 + k)$ *time (if it exists), where n is the number of point-variables (vertices), and k is the number of the inequations.*

Proof (sketch). We can derive an algorithm for computing a solution for a given STN$^{\neq}$ (STP$^{\neq}$) by modifying STP-SOLUTION in the following way:

Algorithm 4: STP$^{\neq}$-SOLUTION
Input: a STP$^{\neq}$ T
Output: a solution for T, if it exists; nil, otherwise

0. Check the consistency of T. If T is not consistent the return nil and stop.
1. Let *M* be the STN$^{\neq}$ computed by the following steps:
   (a) compute the minimal network *V* of the relaxed network of T;
   (b) add to T" the input inequations which are convex relative *T'* and compute the minimal network *T"* of the resulting network;
   (c) add to *T"* the input inequations which are non-convex relative to T";
2. Same as step 2 of STP-CONSISTENCY, except that *6* is defined as:
   - *Min* $\{\delta_{ij}\}$, if at least one interval has both the bounds finite, or the lower bound is finite and the interval is not convex, i.e, it has at least one (finite) point excluded;
   - any finite number otherwise,

   where *i, j* = l..n, $i \neq j$ and δ_{ij} is the length of the *first* convex subinterval of the (possibly non-convex) intervals of M (e.g., for the interval [2, 10] -{3,4} the length of such a subinterval is 1; while for the interval (1,5], the relevant subinterval has length 4).
3. Same as step 3 of STP-SOLUTION, except that *M"* is the minimal network of the STN resulting from *M'* by omitting all the non-convex inequations.
4. Same as step 4 of STP-SOLUTION.

The proof of the correctness of STP-CONSISTENCY is similar to the proof of Theorem 3. Concerning the complexity, note that by Theorem 4 step 0 can be computed in $0(n^3 + k)$ time, step 1 in $0(n^3 + k)$ time, and steps 2-4 in $0(n^3)$ time. □

6 Conclusions

We have presented three new algorithms for computing a solution (scenario) for a set of ORD-Horn relations, for a given STP, and for a STP$^{\neq}$. Also, we have presented an algorithm for checking the consistency of a STP$^{\neq}$, which improves the complexity bound of Koubarakis' method.

Recently Ligozat [1996, personal communication] proved the tractability of the ORD-Horn algebra by using a method for deriving a consistent scenario that is based on iteratively (a) refining a path-consistent network and (b) imposing path-consistency to it. However, he does not give a detailed analysis of his method in terms of time complexity, which appears to be at least one order of magnitude worse than the complexity of our method.

References

[Allen, 1983] J. Allen. Maintaining knowledge about tem-poral intervals. *Communication of the ACM,* 26(l):832-843, 1983.

[Cormen *et al,* 1990] T. Cormen, C. Leiserson, and R. Rivest. *Introduction to Algorithms.* MIT Press, 1990.

[Dechter *et al,* 199l] R. Dechter, I. Meiri, and J. Pearl. Temporal constraint networks. *Artificial Intelligence,* 49:61 95, 1991.

[Gerevini and Cristani, 1995] A. Gerevini and M. Cristani. Reasoning with inequations in temporal constraint networks. In *Proc. of IJCAI-95 Workshop on Spatial and Temporal Reasoning,* pp. 13-23, 1995.

[Gerevini and Schubert, 1995] A. Gerevini and L. Schubert. Efficient Algorithms for Qualitative Reasoning about Time. *Artificial Intelligence* 74:207-248, 1995.

[Henschen and Wos, 1974] L. Henschen and L. Wos. Unit refutations and Horn sets. *Journal of ACM,* 21:590-605, 1974.

[Kautz and Ladkin, 1991] H.A. Kautz and P.B. Ladkin. Integrating metric and qualitative temporal reasoning. In *Proc. of AAAI-91.* The MIT press, 1991.

[Koubarakis, 1992] M. Koubarakis. Dense time and temporal constraints with ≠. In *Proc. of KR&R'92,* pp. 24-35, Cambridge, MA, 1992.

[Koubarakis, 1995] M. Koubarakis. From local to global consistency in temporal constraint networks. In *Proc. of the 1st Int. Conf. on Principle and Practice of Constraint Programming,* 1995. Springer Verlag.

[Ladkin and Maddux, 1988] P.B. Ladkin and R. Maddux. On binary constraint networks. Technical Report KES.U.88.8. Kestrel Institute, Palo Alto, CA, 1988.

[Ligozat, 1996] G. Ligozat. A new proof of the tractability of ORD-Horn Relations. *Proc. of AAAI-96,* pp. 395-401. 1996. AAAI Press/The MIT Press

[Montanari, 1974] U. Montanari. Networks of constraints: Fundamental properties and applications to picture processing. *Information Science,* 7(3):95-132, 1974.

[Nebel and BUirckert, 1995] B. Nebel and H. J. Burckert. Reasoning about temporal relations: A maximal tractable subclass of Allen's interval algebra. *Journal of the* ACM 42(l):43-66, 1995.

[van Beek, 1990] P. van Beek. Reasoning about qualitative temporal information. In *Proc. of AAAI-90,* pp. 728 734, Boston, MA. 1990.

[Vilain and Kautz, 1986] M. Vilain and H.A. Kautz. Constraint propagation algorithms for temporal reasoning. In *Proc. of AAAI-86,* pp. 377-382. 1996.