Deciding consistency of a point-duration network with metric constraints

Isabel Navarrete
Facultad de Informatica
Universidad de Murcia
Campus de Espinardo, Murcia 30.071, Spain
inava@dif.um.es

Abdul Sattar
School of Information Technology, Faculty of Engineering and Information Technology
Griffith University
PMB 50 Gold Coast Mail Centre, Queensland 9726, Australia
a.sattar@gu.edu.au

Roque Marin
Facultad de Informatica
Universidad de Murcia
Campus de Espinardo, Murcia 30.071, Spain
roque@dif.um.es

Abstract

We introduce a new model, MPDN, for quantitative temporal reasoning with points and durations, that supposes an extension of the TCSP formalism and previous point-duration network models. The problem of deciding consistency for a MPDN is shown to be NP-complete. So, we identify a tractable fragment, named simple MPDN, that subsumes the STP model and allows for duration reasoning. Necessary and sufficient conditions for deciding consistency of a simple MPDN are used to design an algorithm for consistency checking, whose time complexity is cubic in the number of variables. This is a significant improvement, not only in computational complexity but also in simplicity, over previous non-specific algorithms that can be applied to solve the consistency problem.

Keywords: Temporal representation and reasoning in AI, temporal constraint reasoning, point and duration reasoning.

1 Introduction

In a wide variety of situations, problem solving tasks require a rather extensive knowledge and reasoning about time. In most applications, knowledge of temporal constraints is expressed in terms of relations between time objects (intervals, points or durations). Several constraint-based systems have been proposed for temporal constraint reasoning, mainly concentrated on two approaches: qualitative formalisms [1, 14] and quantitative or metric models [3]. Later efforts [6, 10] have been made on integrating information between time points and intervals in a single model.

Some researchers have presented systems that support qualitative and/or quantitative constraints between durations [1, 13, 12, 11]. The need for this kind of models is well argued in the literature. In this paper, we present a new point-duration network model with metric constraints, MPDN, which subsumes some previous formalisms for temporal reasoning with points and/or durations [3, 11]. Since the consistency problem for these models is NP-complete, we cannot expect better complexity results for a MPDN. So, we

introduce a tractable fragment, simple MPDN, which is a nice tradeoff between expressive power and time complexity. The consistency checking algorithm we provide for a simple MPDN supposes a significant improvement over previous non-specific algorithms that can be applied to solve the consistency problem [7, 5, 13].

Example 1 *To illustrate the usefulness of the new tractable fragment we adapt the example proposed in [10] and we include additional information about durations of events. The temporal information provided in this story can be managed with a simple MPDN, as we will show later.*

"Bob, Fred and John work for a company that has main office in Los Angeles. It takes John less than 20 minutes and Fred 15-20 minutes to get to work. Today John left home between 7:05-7:10 a.m. and Fred arrived at work between 7:50-7:55 a.m. We know that Fred and John met a traffic light on their way to work. Today Bob left home before 7:45 a.m. and takes 5-10 minutes less than Fred to go to work".

2 Preliminaries

A *metric constraint* (or quantitative constraint) is represented by a set of real intervals $C = \{I_1, \ldots, I_k\}$. A *unary metric* constraint C_i restricts the domain of variable x_i to the given set of intervals, while a *binary metric* constraint $C_{i,j}$ restricts the feasible values for the time-distance $x_j - x_i$.

A *TCSP* [3] is a binary network involving a set of point variables and sets of unary and binary metric constraints among them. A special case is an *STP* where each constraint is given by a single interval. We will use the *metric algebra* [3], that is provided with operations of set intersection, inverse $(^{-1})$ and composition $(\otimes)$ of metric constraints. Given two quantitative constraints C and C', the composition $C \otimes C'$ can be computed as the union of pair-wise sum of intervals, that is:

$$C \otimes C' = \bigcup_{I_j \in C, I_k \in C'} I_j + I_k =$$
$$[a_j, b_j] + [a_k, b_k] = [a_j + a_k, b_j + b_k]$$

The inverse C^{-1} is obtained as $C^{-1} = \left\{ I_j^{-1} \mid I_j \in C \right\}$, where the inverse of an interval $[a, b]$ is the interval $[-b, -a]$.

Definition 1 *A metric point-duration network, MPDN, is an structure* $\Sigma_{PD} = \langle N_P, N_D, Rel(P, D) \rangle$ *formed by two TCSPs, N_P and N_D, and a set of ternary constraints $Rel(P, D)$ relating points and durations, where:*

- *N_P is determined by a set $P = \{p_1, \ldots, p_n\}$ of time-point variables that take values over $\mathbb{R}_0^+$, and a set of unary and binary metric constraints between points.*

- *N_D is given by a set $D = \{d_{ij} \mid p_i, p_j \in P\}$ of duration variables over $\mathbb{R}_0^+$ and a set of unary and binary metric constraints between durations.*

- *$Rel(P, D)$ is given by triplets of real values, (P_i, P_j, D_{ij}), for points and durations that satisfy the Euclidean distance equation $d_{ij} = |p_i - p_j|$, for echa duration d_{ij}.*

A MPDN with n points and d durations is *consistent* if at least one *solution* $S = (A_P, A_D)$ exists, where A_P is a n-tuple of pairs, where each pair $< p_i, P_i >$ denotes the assignment of a value P_i to point variable p_j, and A_D is a d-tuple of pairs $< d_{ij}, D_{ij} >$, where a value is assigned to each duration variable, so that all unary, binary and ternary constraints are satisfied. A constraint is *feasible* is there is a solution that satisfies this constraint. The *minimal MDPN* [3] equivalent to a given one is represented by all feasible unary and binary constraints, what means that these constraints are as explicit as possible.

Definition 2 *A* simple MPDN *is a MPDN such that each network N_P and N_D represents an STP, and for every duration d_{ij} it must be $p_i \leq p_j$, that is $C_{i,j} \subseteq [0, \infty)$. This way $Rel(P, D)$ is obtained upon the linear equation $d_{ij} = p_j - p_i$ for each duration d_{ij}.*

In order to deal with consistency and other reasoning tasks, it is useful to **represent a MPDN** (or simple MPDN) by means of two directed constraint graphs, $G_P = (V_P, E_P)$ and $G_D = (V_D, E_D)$, that explicitly show binary constraints affecting points and durations, respectively. One node represents a point or duration variable, that may be labelled with its unary constraint. Each arc is labelled with the corresponding binary constraint.

Example 2 *The temporal information given in the story of example 1 can be managed with a simple MPDN. Let $b^-, b^+, f^-, f^+, j^-, j^+$ denote the time points that Bob, Fred and John, respectively, leave home and arrive at the office. The constraint graphs G_P and G_D depicted in figure 1, show the unary and binary metric constraints between points and durations that can be extracted from the story. All arcs are supposed to be labelled with $(0, \infty)$, which is equivalent to the qualitative relation $<$ [10]. All times in G_p are relative to the "beginning of the world", chosen at 7:00 a.m. For instance, from the given information that Fred arrives at work between 7:50-7:55 a.m., the domain of f^+ is restricted to the time interval (50,55). The duration of Fred going to work is also limited to (15,20). The incomplete qualitative information that "Fred and John met at a traffic light on their way to work" can be interpreted as the IA-relation [1] {start, started-by, during, contain, finish, finished-by, overlapped, overlapped-by, equal} between the two interval events of can be represented by a conjunction of PA-relations (in this case $<$-relation or equivalently $(0, \infty)$) between the endpoints of the intervals.*
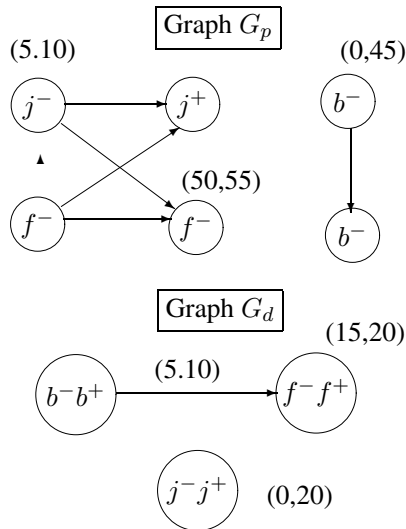
Figure 1. Constraint graphs for example 1

A unary constraint C_i for a point p_i can be turned into a binary constraint $C_{0,i}$ between the *beginning point*

p_0 and p_i. Similarly, a unary constraint C_{ij} for a duration d_{ij} can be expressed as a binary constraint $C_{00,ij}$ between the *null duration* d_{00} and d_{ij}. Notice that the domain (i.e., unary metric constraint) C_{ij} for duration d_{ij} can be considered as a binary metric constraint $C_{i,j}$ between points p_i and p_j, since $d_{ij} = p_j - p_i$. For instance, in the graph G_p of figure 1, binary constraint between points f^- and f^+ may be updated to $C_{f^-, f^+} = (15, 20)$, since $C_{f^- f^+} = (15, 20)$. Now we define two STPs associated to binary constraints for points and durations in a simple MPDN.

Definition 3 *Let Σ_{SMPD} be a simple MPDN. The* **STP-P** *of Σ_{SMPD} is a constraint graph whose nodes represent the points in N_P plus the beginning point p_0, and for every pair of points p_i, p_j there is an arc $i \xrightarrow{T_{i,j}} j$, such that,*

- $T_{i,j} = C_{i,j} \cap C_{ij}$, if $d_{ij} \in D$
- $T_{i,j} = C_j$, if $i = 0$
- $T_{i,j} = C_{i,j}$, otherwise.

Definition 4 *Let Σ_{SMPD} be a simple MPDN. The* **STP-D** *of Σ_{SMPD} is a constraint graph whose nodes represent the durations in N_D plus the null-duration d_{00}, and for every pair of durations d_{ij}, d_{km} there is an arc $ij \xrightarrow{T_{ij,km}} km$, such that,*

- $T_{ij,km} = C_{ij,km}$ if $ij \neq 00$.
- $T_{ij,km} = C_{km} \cap C_{k,m}$ if $ij = 00$.

3 Tractability of the simple MPDN fragment

The main temporal reasoning task within a temporal constraint model is determining the consistency or satisfiability of the network. The consistency problem for either PDN and APDN point-duration formalisms is NP-complete [11] and tractable classes of these models have been identified, but they have a limited expressive power. In this work we identify a new tractable and quite expressive fragment for handling with metric temporal information between points and durations, as we show in the next theorem.

Theorem 1 *The consistency of a simple MPDN can be decided in polynomial time.*

Proof.- It is easy to see that all the constraints in a simple MPDN can be turn into *Horn constraints* [7, 5], i.e., a set of disjunctions of weak linear inequalities and linear inequations, with at most one inequality per disjunction. Hence, Koubarakis' CONSISTENCY algorithm [7], which is polynomial, can be used in the simple MPDN fragment. $\square$

The major drawback of Koubarakis' algorithm is its high complexity. It is based on the application of a polynomial linear programming method with high complexity[1]. Hence, some specific and efficient method is needed for our simple MPDN fragment. For this reason we are going to investigate under which conditions one has the certainty that a simple MPDN is consistent.

Suppose we have three p_i, p_j, p_k points and pairs of durations in which these points are involved. The following equations show the influence of binary constraints between points over binary constraints between durations and vice versa:

$$d_{ik} - d_{ij} = (p_k - p_i) - (p_j - p_i) = p_k - p_j \quad (1)$$
$$d_{kj} - d_{ij} = (p_j - p_k) - (p_j - p_i) = p_i - p_k \quad (2)$$

Definition 5 *Upon equations (1),(2) we say that a simple MPDN satisfy **PD3-conditions** if and only if,*

$$\forall d_{ij}, d_{ik} \in D : T_{ij,ik} = T_{j,k}$$
$$\forall d_{ij}, d_{kj} \in D : T_{ij,kj} = T_{k,i}$$

where each binary metric constraint corresponds to the constraint given by the STP-P or STP-D of the simple MPDN.

Suppose now we have four points p_i, p_j, p_k, p_m and pairs of durations in which these points appear. The influence of binary constraints between points over binary constraints between durations and vice versa is shown in the following equations:

$$p_m - p_j = (d_{km} - d_{ij}) + (p_k - p_i) \quad (3)$$
$$p_k - p_i = (d_{ij} - d_{km}) + (p_m - p_j) \quad (4)$$
$$d_{km} - d_{ij} = (p_i - p_k) + (p_m - p_j) \quad (5)$$
$$d_{km} - d_{ij} = (p_m - p_j) - (p_k - p_i) = d_{jm} - d_{ik} \quad (6)$$
$$d_{km} - d_{ij} = (p_i - p_k) - (p_j - p_m) = d_{ki} - d_{mj} \quad (7)$$

---

[1]A linear programming problem in n variables can be solved in $O(n^5 \log T)$ arithmetic operations on numbers with $O(n^3 \log T)$ digits, where T is the maximum absolute value of the entries.

Definition 6 *Upon equations (3)-(7) we say that a simple MPDN satisfy **PD4-conditions** if and only if,*

$$\forall p_i, p_j, p_k, p_m \in P, \ \forall d_{ij}, d_{km} \in D :$$
$$\begin{cases} T_{j,m} \subseteq T_{ij,km} \otimes T_{i,k} \\ T_{i,k} \subseteq T_{km,ij} \otimes T_{j,m} \\ T_{ij,km} \subseteq T_{k,i} \otimes T_{j,m} \end{cases}$$
$$\forall d_{ij}, d_{km}, d_{ik}, d_{jm} \in D : T_{ij,km} = T_{ik,jm}$$
$$\forall d_{ij}, d_{km}, d_{ki}, d_{mj} \in D : T_{ij,km} = T_{mj,ki}$$

We refer to PD3 and PD4-conditions altogether as **PD-conditions**.

Theorem 2 *A simple MPDN is consistent if STP-P is path consistent [9], STP-D is path consistent and PD-conditions are satisfied.*

Proof.- Let $\Sigma_{SMPD} = < N_P, N_D, Rel(P, D) >$ be a simple MPDN. Satisfiability of binary constraints alone is guaranteed since both binary networks N_P and N_D, represented as STP networks, are path consistent and minimal and so consistent [3]. We show that ternary constraints in $Rel(P, D)$ are also satisfied if all PD-conditions hold. Suppose Σ_{SMPD} is inconsistent, but STP-P and STP-D are path consistent. The only source of inconsistency is due to the influence of binary constraints between points over binary constraints between durations and vice versa. This influence has been shown in the equations (1)-(7), upon which PD3 and PD4-conditions have been obtained. Then, if Σ_{SMPD} is inconsistent this is because some PD-condition is not satisfied. For instance, if $C_{j,k} = [-10, -8]$ and $C_{ij,ik} = [7, 12]$, the the simple MPDN is inconsistent because there is no assignment for points and durations such that ternary constraints $d_{ij} = p_j - p_i$ and $d_{ik} = p_k - p_i$ are satisfied. Indeed, it must be $7 \leq d_{ik} - d_{ij} \leq 12$, but $7 \leq (p_k - p_i) - (p_j - p_i) \leq 12$, so that $7 \leq p_k - p_j \leq 12$, which is not consistent with $C_{j,k} = [-10, -8]$. The inconsistency is due to $C_{ij,ik} \cap C_{j,k} = \varnothing$, so that one PD3-condition is not satisfied. All the inconsistencies we can find in a simple MPDN with path consistent STP-graphs are due to empty intersections in PD3 or PD4 conditions. Otherwise the network would be consistent. In summary, conditions of theorem **??** are **sufficient conditions** to ensure the consistency of simple MPDN. But **necessary conditions** are given by the fact that the STP-P and STP-D must be consistent and each intersection that substitutes $=$ or $\subseteq$ in PD-conditions must be non empty. $\square$

4 Consistency checking

We present now a consistency algorithm CONS-SMPDN (figure 2) that takes as input a simple MDPN and return TRUE when the network is consistent and FALSE otherwise. This algorithm is based on Mackworth's PC-2 algorithm [9] for achieving path consistency. Our goal is to accomplish, if possible, path consistency in the STP-P and STP-D graphs at the same time that satisfiability of PD-conditions is guaranteed.

Every different triple of point or duration-nodes, representing paths of length two, are added to the sets Q_P and Q_D, respectively. For every triple we check if the label (constraint) of one of the arcs changes as a product of a composition operation or as a consequence of a call to procedures PD3 (figure 3) or PD4 (figure 4), that are used to ensure the satisfiability of PD-conditions. When a label of an arc changes, its effect must be propagated, so we use a function REL-PATHS that take an arc (between point or durations) and returns a set of triples representing all paths of length two in which the arc participates.

```
 1. repeat
 2. while Q_P ≠ ∅
 3.    delete path (i, j, k) from Q_P;
 4.    t ← T_{i,k} ∩ (T_{i,j} ⊗ T_{j,k}); PD3 (i, j, k, );
 5.    if T_{i,k} ≠ t then
 6.       T_{i,k} ← t; if T_{i,k} = ∅ then exit (FALSE);
 7.       Q_P ← Q_P ∪ REL-PATHS (i, k, Q_P); PD3 (i, j, k);
 8.       if d_{ik} ∈ D then Q_D ← Q_D ∪ REL-PATHS (00, ik, Q_D);
 9. while Q_D ≠ ∅
10.    delete path (ij, pq, km) from Q_D;
11.    t ← T_{ij,km} ∩ (T_{ij,pq} ⊗ T_{pq,km})
12.    if ij, km ≠ 00 then if i = k then PD3 (j, i, m)
13.    else if j = m then PD3 (i, j, k) else PD4 (ij, km);
14.    if T_{ij,km} ≠ t then
15.       T_{ij,km} ← t; if T_{ij,km} = ∅ then exit (FALSE);
16.       Q_D ← Q_D ∪ REL-PATHS (ij, km, Q_D);
17.       if ij = 00 then Q_P ← Q_P ∪ REL-PATHS (k, m, Q_P);
18.       if km = 00 then Q_P ← Q_P ∪ REL-PATHS (i, j, Q_P);
19.       if ij, km ≠ 00 then if i = k then PD3 (j, i, m, )
20.       else if j = m then PD3 (i, j, k) else PD4 (ij, km);
21. until (Q_P = ∅) ∧ (Q_D = ∅);
```

Figure 2. Algorithm CONS-SMPDN (Σ_{SMPD})

```
1. if d_{ji}, d_{jk} ∈ D then
2.    t ← T_{ji,jk} ∩ T_{i,k}; if t = ∅ then exit (FALSE);
3.    if T_{i,k} ≠ t then
4.       T_{i,k} ← t; Q_P ← Q_P ∪ RELATED-PATHS (i, k, Q_P);
5.       if d_{ik} ∈ D then
6.          Q_D ← Q_D ∪ RELATED-PATHS (00, ik, Q_D);
7.    if T_{ji,jk} ≠ t then
8.       T_{ji,jk} ← t; Q_D ← Q_D ∪ RELATED-PATHS (ji, jk, Q_D);
9. end-if;
```

Figure 3. Procedure PD3 (i, j, k)

Theorem 3 *The algorithm* CONS-SMPDN *correctly checks for consistency in a simple MPDN in* $O(n^3 + d^3)$, *where* n *is the number of points,* d *is the number of durations.*

Proof.- The correctness of the algorithm follows from the correctness of PC-2 algorithm for deciding consistency in the STP model [3] and necessary and sufficient conditions to ensure consistency of a simple MPDN (see theorem 2). When the algorithm returns FALSE then the simple MPDN is inconsistent since necessary conditions for consistency are not satisfied (some constraint becomes empty). When the algorithm returns TRUE the network is consistent, since sufficient conditions for consistency are fulfilled.

For the analysis of the time complexity we take as a reference the time complexity of PC-2 applied to an STP, which is $O(n^3)$ for n points [3]. In addition to the paths processed by PC-2, either in STP-P or STP-D ($O(n^3 + d^3)$ in total), here when a binary constraint $T_{i,j}$ changes then at most $O(d)$ paths are added to Q_D (if $d_{ij} \in D$) and when a binary constraint between durations changes then at most $O(n)$ paths are added to Q_P (for constraints of the form $T_{00,ij}$). In total, no more than $O(n^3 + n^2 \times d + d^2 \times n + d^3) = O(n^3 + d^3)$ paths are processed. $\square$

Since we know how to solve a simple MPDN, a backtracking algorithm could be devised for solving a MPDN following the same idea proposed for a TCSP [3], where, in addition, the network for durations must be considered now. The consistency algorithm for a MPDN must find a consistent simple MPDN extracted from the input network. If such a consistent subnetwork cannot be found the MPDN is inconsistent.

1. if $d_{ik}, d_{jm} \in D$ then
2. $t \leftarrow T_{ij,km} \cap T_{ik,jm}$; if $t = \varnothing$ then $exit$ (FALSE);
3.   if $T_{ij,km} \neq t$ then
4. $T_{ij,km} \leftarrow t$;
5. $Q_D \leftarrow Q_D \cup$ REL-PATHS (ij, km, Q_D);
6.   if $T_{ik,jm} \neq t$ then
7. $T_{ik,jm} \leftarrow t$;
8. $Q_D \leftarrow Q_D \cup$ REL-PATHS (ik, jm, Q_D);
9. $t \leftarrow T_{j,m} \cap (T_{ij,km} \otimes T_{i,k})$; if $t = \varnothing$ then $exit$ (FALSE);
10. if $T_{j,m} \neq t$ then
11. $T_{j,m} \leftarrow t$;
12. $Q_P \leftarrow Q_P \cup$ REL-PATHS (j, m, Q_P);
13.   if $d_{jm} \in D$ then
14. $Q_D \leftarrow Q_D \cup$ REL-PATHS $(00, jm, Q_D)$;
15. $t \leftarrow T_{i,k} \cap (T_{km,ij} \otimes T_{j,m})$; if $t = \varnothing$ then $exit$ (FALSE);
16. if $T_{i,k} \neq t$ then
17. $T_{i,k} \leftarrow t$;
18. $Q_P \leftarrow Q_P \cup$ REL-PATHS (i, k, Q_P);
19.   if $d_{ik} \in D$ then
20. $Q_D \leftarrow Q_D \cup$ REL-PATHS $(00, ik, Q_D)$;
21. $t \leftarrow T_{ij,km} \cap (T_{k,i} \otimes T_{j,m})$; if $t = \varnothing$ then $exit$ (FALSE);
22. if $T_{ij,km} \neq t$ then
23. $T_{ij,km} \leftarrow t$;
24. $Q_D \leftarrow Q_D \cup$ REL-PATHS (ij, km, Q_D);
25. end-if;

Figure 4. Procedure PD4 (ij, km)

Theorem 4 *The consistency problem for a metric point-duration network is NP-complete.*

Proof.- A non-deterministic algorithm can check for consistency in a MPDN in polynomial time, using the algorithm CONS-SMPDN with each simple subnetwork extracted from the MPDN. So, solving a MPDN is NP. Moreover, it is NP-complete because deciding consistency of a TCSP is NP-complete [3] and this model can be considered as a special case of a MPDN with no duration variables. □

Example 3 *The story of example 1 continues...* "Bob, Fred and John's boss has been found murdered this morning at office. The police inspector has been asking some questions and he has found out that:

- A neighbor heard a shot gun at 7:25 a.m.

- John declared to arrive at the office at 7:35 a.m. His boss was already dead and Bob also was at the office.

- Bob said he arrived at 7:40. John was there and his boss was lying on the floor.

- Fred said he arrived at 7:53 and, apart from the dead body, Bob and John were there with angry faces".

What can be said about these declarations above? Applying our algorithm CONS-SMPDN *to the corresponding simple MPDN (see example 2) give rise to the minimal STP-D depicted in figure 5. Part of the minimal STP-P that is relevant at this point is also shown in the figure. Fred certainly was not at the office at 7:25, when the crime took place, but John and Bob could have been there. Bob and John's declarations are inconsistent, so one of them may be the murder. The inspector has found out (with the help of the minimal STP-P) that it was impossible that John arrived at the office at 7:35 a.m. John lies, but is he the murder? This is another question.*
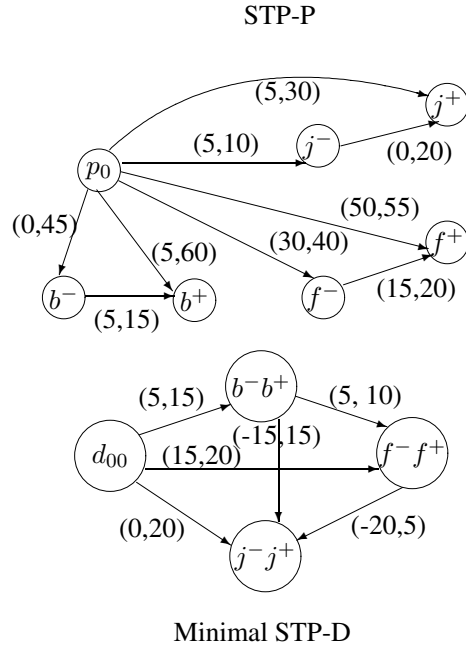
STP-P

Minimal STP-D

Figure 5. Minimal STP-D and STP-P

5 Discussion

Two point-duration network based models has been previously identified [11], PDN and APDN. The first one consists on two point algebra (PA) networks [14] related by ternary constraint as in $Rel(P, D)$ and the second one is given by two augmented PA networks [10] with ternary constraints. The APDN model is an extension of a TCSP which allows for duration reasoning [11]. Here we further extend the APDN since point and duration variables are now related by metric constrains, and every qualitative relation in the PA can be turned into a metric constraint [10].

Notice that each duration variable d_{ij} in the MPDN model represents the elapsed time between two temporal points p_i and p_j, but it does not supposes anything about the relative position of these points. This is because we use the Euclidean distance $d_{ij} = |p_i - p_j|$ to relate points with durations. It is worth noting that the these constraints cannot be represented with disjunctive linear relations (DLRs) [5, 7], which is known to be a very expressive formalism to deal with temporal constraints. The tractable fragment, simple MPDN, we identify here is not very restrictive since condition $p_i \leq p_j$ for a duration d_{ij} is not so strong. This is similar to say, in the interval algebra context [1], that the start point of a time interval is before its end point.

A simple MPDN can be represented as a pre-convex PIDN [13]. In this fragment of the PIDN model, checking for consistency requires a 4-consistency algorithm which is at least $O(t^4)$ [2], being t the number of temporal objects (points or intervals). New composition, inverse and intersection operations must be used with de consistency algorithm, which is not shown in [13]. Our consistency checking algorithm for a simple MPDN is an easy path consistency like algorithm and it improves the complexity of the algorithm for the pre-convex PIDN fragment by at least a linear factor– a significant speed up. This is because, in order to represent p points and d durations in a pre-convex PIDN, one needs $O(max(p \div 2, d))$ objects, so that when $d > p \div 2$ our algorithm is always better and when $d \leq p \div 2$ our algorithm is better for $p > 4$.

6 Conclusion

We have introduced a new model, MPDN, for metric temporal reasoning with points and durations, that supposes an extension of the TCSP formalism [3] and previous point-duration network models [11]. The problem of deciding consistency for a MPDN is shown to be NP-complete.

We have identified a tractable fragment of the MPDN model which reflects a good tradeoff between expressive power and complexity. This class, simple MPDN, allows representing points and durations as temporal objects and can manage with simple metric constraints. We have investigated necessary and sufficient conditions to ensure the consistency of a simple MPDN and, using these conditions, a constraint propagation algorithm for consistency checking has been designed. With our consistency algorithm, non-binary constraints (e.g., ternary relations and 4-ary implicit constraints among points involved in a binary constraint between duration) are managed with a formalism that integrates two well studied binary constraint networks. The time complexity of the algorithm is cubic in the number of variables what supposes a nice improvement, not only in computational complexity, but also in simplicity, over previous non-specific algorithms that can be applied to solve the consistency problem.

An easy extension to the simple MPDN fragment is to cope with binary constraints expressed as intervals with holes, so that we can obtain algorithms for consistency checking and finding the minimal network following the ideas presented in [4, 8]. We are also working in an algorithm for finding a solution to a simple MPDN that it may be useful for several applications, specially for those where the STP fragment [3] has been used.

Acknowledgments

IEEE
COMPUTER
SOCIETY

References

[1] J. Allen. Maintaining knowledge about temporal intervals. *Communications of the ACM*, 26(11):832–843, 1983.

[2] M. Cooper. An optimal k-consistency algorithm. *Artificial Intelligence*, 41:89–95, 1989/90.

[3] R. Dechter, I. Meiri, and J. Pearl. Temporal constraint networks. *Artificial Intelligence*, 49:61–95, 1991.

[4] A. Gerevini and M. Cristani. On finding a solution in temporal constraint satisfaction problems. In *Proceedings of IJCAI-97*, pages 1460–1465, 1997.

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

[6] H. Kautz and P. Ladkin. Integrating metric and qualitative temporal reasoning. In *Proceedings of AAAI-91*, pages 241–246, 1991.

[7] M. Koubarakis. Tractable disjunctions of linear constraints. *Lecture Notes in Computer Science*, 1118:297–307, 1996.

[8] M. Koubarakis. From local to global consistency in temporal constraint networks. *Theoretical Computer Science*, 173(1):89–112, 1997.

[9] A. Mackworth and E. Freuder. The complexity of some polynomial network consistency algorithms for constraint satisfaction problems. *Artificial Intelligence*, 25:65–74, 1985.

[10] I. Meiri. Combining qualitative and quantitative constraints in temporal reasoning. *Artificial Intelligence*, 87:343–385, 1996.

[11] I. Navarrete, A. Sattar, R. Wetprasit, and R. Marin. On point-duration networks for temporal reasoning. *Artificial Intelligence*, 140:39–70, 2002.

[12] A. Pujari, G. Kumari, and A. Sattar. Indu-qualitative-interval-duration network. In *Australian Joint Conference on Artificial Intelligence*, pages 291–303, 1999.

[13] A. Pujari and A. Sattar. A new framework for reasoning about points, intervals and durations. In *Proceedings of IJCAI-99*, pages 1259–1264, 1999.

[14] M. Vilain and H. Kautz. Constraint propagation algorithms for temporal reasoning. In *Proceedings of the National Conference on Artificial Intelligence (AAAI-86)*, pages 377–382, 1986.