A Method for Metric Temporal Reasoning

Mathias Broxvall *

Department of Computer and Information Science
Linköpings Universitet
S-581 83 Linköping, Sweden
matbr@ida.liu.se

Abstract

Several methods for temporal reasoning with metric time have been suggested—for instance, Horn Disjunctive Linear Relations (Horn DLRs). However, it has been noted that implementing this algorithm is non-trivial since it builds on fairly complicated polynomial-time algorithms for linear programming. Instead, an alternative approach which augments Allen's interval algebra with a Simple Temporal Problem (STP) has been suggested (Condotta, 2000). In this paper, we present a new point-based approach STP* for reasoning about metric temporal constraints. STP* subsumes the tractable preconvex fragment of the augmented interval algebra and can be viewed as a slightly restricted version of Horn DLRs. We give an easily implementable algorithm for deciding satisfiability of STP* and demonstrate experimentally its efficiency. We also give a method for finding solutions to consistent STP* problem instances.

Introduction

Temporal reasoning in various forms has long been an important task in many areas of AI. Examples of tasks that have been studied is that of deciding consistency or finding a solution of a set of constraints over temporal variables. Typically these variables represents time points or time intervals over a linear time domain. The constraints imposed on these variables can either be of a purely qualitative nature, stating relations such as precedence or equality, or of a more quantitative nature. Examples of the later case would be to state that a certain time point occurs at least a given amount of time units before another or that an interval is of a certain length.

The time complexity for qualitative reasoning is fairly well understood. Reasoning about intervals is commonly done using Allen's interval algebra (Allen 1983) for which the satisfiability problem is NP-complete in the general case but for which several tractable fragments have been identified. One of the more useful tractable fragments is the ORD-Horn fragment (Nebel & Bürckert 1995) which consists of the 868 interval relations which can be expressed by conjunctions of constraints of the form $x \;R\; y \vee z \neq w$ where x, y, z, w denotes endpoints of the intervals and $R \in \{\leq, \neq\}$. Note that x, y or z, w may denote the same endpoints for non-disjunctive constraints such as the interval constraint "before or meets" which can be written as $x^+ \leq y^- \vee x^+ \neq x^+$. The ORD-Horn fragment is also known as the set of preconvex interval algebra relations. Qualitative reasoning on time points is done using the point algebra and satisfiability can easily be decided using path consistency.

A number of methods has been proposed for reasoning about metric time. One of these is *Horn Disjunctive Linear Relations* (Jonsson & Bckstrm 1998; Koubarakis 2001), Horn DLRs for short. This is a temporal constraint formalism for which satisfiability can be checked in polynomial time. The expressibility of Horn DLRs subsumes the ORD-Horn algebra and most of the other tractable formalisms for temporal reasoning and satisfiability is decided using a linear programming approach. As a consequence, the algorithm for deciding satisfiability is not trivial to implement and special considerations need to be taken to the numerical precision during calculations. These problems have been pointed out in a number of papers, cf. (Condotta 2000; Pujari & Sattar 1999). For a good presentation of the polynomial time algorithms for linear programming and the numerical issues involved see eg. (Papadimitriou & Steiglitz 1982). Another slightly more efficient algorithm solving the linear programming problem has been presented by Vaidya (1987).

A method proposed by Condotta (2000) handles metric time by augmenting Allen's interval algebra and the rectangle algebra (Güsgen 1989) with quantitative STP constraints on the endpoints. Although this method yields an algorithm which is simpler to implement than the Horn DLR approach it lacks the expressibility of Horn DLRs and has the rather high time complexity of $O(n^5)$.

In this paper we present yet another approach for handling metric time called STP* which has an expressibility subsuming that of the tractable ORD-Horn fragment of the augmented interval (and rectangle) algebra but slightly more restricted than Horn DLRs. By restricting ourselves to using this method we get a very simple algorithm for deciding satisfiability that has a low practical time complexity and a good expressibility. We recall that the Simple Temporal Problem (STP) allows metric constraints of the form

*This research has been supported by the ECSEL graduate student program.

1 Algorithm: STP-1
2 Input: A set C of STP^* clauses over the set of variables $1 \dots n$.
3 Let M be the $n \times n$ matrix such that $M[k][l] = 0$ if $k = l$ and ∞ otherwise .
4 **repeat**
5 $\quad changed \leftarrow false$
6 $\quad$ update M to contain the shortest path between every pair of variables.
7 $\quad$ **if** $\exists x : M[x][x] < 0$ **then reject**
8 $\quad$ **for** every clause $c = x_1 + r_1 \leq y_1 \vee x_2 + r_2 \neq y_2 \vee \cdots \vee x_m + r_m \neq y_m$ in C **do**
9 $\quad\quad$ **if** $r_1 < M[x_1][y_1] \wedge \neg\exists i > 1 : M[x_i][y_i] \neq r_i \vee M[y_i][x_i] \neq r_i$ **then**
10 $\quad\quad\quad M[x_1][y_1] \leftarrow r_1$
11 $\quad\quad\quad changed \leftarrow true$
12 $\quad\quad\quad$ remove c from C
13 $\quad\quad$ **end**
14 **until** $changed = false$
15 **accept**

Figure 1: The algorithm for solving STP* problem instances

$x + r \leq y$ where x, y represents point variables and r is a real constant. The satisfiability problem for STPs and STPs augmented with disequality constraints (STP$^{\neq}$) relations has been well investigated previously (Dechter, Meiri, & Pearl 1991; Gerevini & Cristani 1997) and efficient algorithms has been identified for deciding satisfiability. By basing our approach on STPs and allowing disjunctions of disequality we get an expressive formalism that retain most of the simplicity and efficiency of the satisfiability algorithms for STP and STP$^{\neq}$. As we will see later in this paper the practical cost of adding metric information is relatively small when we use this method.

We continue this paper by recalling the definitions of STP$^{\neq}$, presenting STP* and an algorithm for solving the satisfiability problem in the next section. In the following section we investigate the expressibility of STP* and show that the tractable preconvex fragment of the augmented interval algebra can be expressed in terms of STP* constraints and that certain rectangle algebra relations also can be expressed in STP*. After this we continue with some studies and comparisons of an actual implementation of this algorithm to other approaches. Finally, in the last section we have some concluding remarks and open questions.

Simple Temporal Problem

We begin this section by giving the definition of STP$^{\neq}$ introduced by Gerevini and Cristani (1997) and recall some of their previous results. This is in turn followed by the necessary definitions for extending STP$^{\neq}$ with disjunctions of disequality, yielding STP*, and an algorithm for deciding satisfiability of STP* problem instances.

Definition 1 An STP$^{\neq}$ problem instance is a tuple $\langle V, C\rangle$ of a set of variables V and constraints C such that all constraints in C is of the form $x + r \leq y$ or $x + r \neq y$ for $x, y \in V$ and $r \in \mathtt{R}$. The satisfiability problem for STP$^{\neq}$ is that of finding a mapping from the variables onto $\mathtt{R}$ satisfying all the constraints C.

We say that an STP$^{\neq}$ problem instance $\mathcal{T}$ entails $x+d = y$ iff there exists no solution for $\mathcal{T}$ such that $x + d \neq y$. An STP$^{\neq}$ problem instance can also be considered as a labeled graph over its variables with an edge labeled r between x, y for each constraint $x - r \leq y$. This graph is called the distance graph of the instance. For STP$^{\neq}$ we have the following result proven by Gerevini and Cristani (1997) which is helpful for deciding satisfiability of STP$^{\neq}$ problem instances. This will later be used in the proofs for the algorithm deciding satisfiability of STP* instances.

Lemma 2 An STP$^{\neq}$ problem instance $\mathcal{T}$ is consistent iff $\mathcal{T}$ does not have negative cycles in its distance graph, and it does not entail $w + d = v$ for any inequation $w + d \neq v$ in $\mathcal{T}$.

We are now ready to define the STP* point algebra and the satisfiability problem for it. As we will see, STP* and STP$^{\neq}$ are closely related and the algorithm for deciding satisfiability of STP* problem instances resembles that for deciding satisfiability of STP$^{\neq}$ instances given by Gerevini and Cristani (1997).

Definition 3 An STP* problem instance is a tuple $\langle V, C\rangle$ of a set of variables V and constraints C such that all constraints in C is of the form $x_1 + r_1 \leq y_1 \vee x_2 + r_2 \neq y_2 \vee \cdots \vee x_n + r_n \neq y_n$ for $x, y \in V, r \in \mathtt{R}$ and $n \geq 1$. The satisfiability problem for STP* is that of finding a mapping from the variables onto $\mathtt{R}$ satisfying all the constraints C.

The maximum clause size of a problem instance is the largest n for all the constraints in it. Note that the definition allows simple $x + r \leq y$ constraints by having $n = 1$. Since STP* is a restriction of the Horn-DLR framework satisfiability could in principle be decided using the independence algorithm (Cohen *et al.* 2000) and a solver for STP$^{\neq}$. Indeed, the algorithm we present here is largely based on the independence algorithm but by using Lemma 2 we have enhanced the algorithm so that a solution is created incrementally and a complete call to an underlying STP$^{\neq}$ solver can be avoided in each iteration which makes the algorithm more efficient when a shortest path algorithm which handles incremental changes is used in step 6 of the algorithm. In our implementation a modified Floyd-Warshall (Cormen, Leiserson, & Rivest 1997) that keep tracks of modified variables is used for this purpose. We can now prove that the algorithm given in Figure 1 only accepts satisfiable problem instances.

Theorem 4 Algorithm STP-1 accepts only satisfiable STP* problem instances

Proof: Let Π be an STP* problem instance that is accepted by STP-1 and let M, C be the adjacency matrix and the set of clauses left when the algorithm accepts. We construct an STP$^{\neq}$ problem instance Π' from Π by having the constraints $x - M[x][y] \leq y$ for all x, y such that $M[x][y] < \infty$. For each clause $c = x_1 + r_1 \leq y_1 \vee x_2 + r_2 \neq y_2 \vee \cdots \vee x_m + r_m \neq y_m$ in C we add the constraint $x_i + r_i \neq y_i$ for some i such that $M[x_i][y_i] \neq r_i$ or $M[y_i][x_i] \neq r_i$.

Obviously, Π is satisfiable if Π' is satisfiable. We note that Π' contains no negative cycles and that for each constraint $x + r \neq y$ we have that Π does not entail $x + r = y$. By lemma 2 we have that Π' is satisfiable and thus Π is also satisfiable. □

We continue by proving that the algorithm only rejects unsatisfiable STP* problem instances.

Theorem 5 Algorithm STP-1 rejects only unsatisfiable STP* problem instances.

Proof: Let Π be an STP* problem instance that is rejected by the algorithm and let M, C be the adjacency matrix and the set of clauses left when the algorithm rejects. Assume that Π is satisfiable. It is then possible to construct an STP$^{\neq}$ problem instance Π' by selecting one term of each clause in Π such that Π' is satisfiable. From lemma 2 we know that Π' contains no negative cycles and that for each $x + r \neq y$ constraint we have that Π' does not entail $x + r = y$.

We say that M contain a constraint $x - r \leq y$ if $M[x][y] = r$ and $r < \infty$ and we show that M only contains constraints that are present in Π' by induction on the number of constraints in M. The base case of zero constraints is trivially true. For the inductive case assume that this holds for the first n constraints added to M by the algorithm and let $c = x_1 + r_1 \leq y_1 \vee x_2 + r_2 \neq y_2 \vee \cdots \vee x_m + r_m \neq y_m$ be the clause containing the $(n+1)$'th constraint $x_1 + r_1 \leq y_1$ added. Since we have $\neg\exists i > 2 : M[x_i][y_i] \neq r_i \vee M[y_i][x_i] \neq r_i$ and all constraints in M is also present in Π' we see that $x_i + r_i = y_i$ is entailed by Π' for all $i > 2$. Thus, Π' must also contain the constraint $x_1 + r_1 \leq y_1$ added by the algorithm to M.

Now, since M only contains constraints in Π' and the algorithm rejects we see that M and Π' has a negative cycle. Which contradicts lemma 2. Thus, Π is unsatisfiable. □

Next, we note that the algorithm runs in $O(|C|(|V|^3 + |C|t))$ time for problem instances $\langle V, C\rangle$ with maximum clause size t since the shortest path problem can be solved in $O(n^3)$ using eg. the Floyd-Warshall algorithm (Cormen, Leiserson, & Rivest 1997). For converted interval and rectangle problem instances our algorithm thus takes $O(n^5)$ time where n is the number of variables in the original problem instance. Thus, our algorithm has the same asymptotical time complexity as Condottas algorithm for the preconvex fragment of the augmented interval algebra but with with a greater expressibility.

For some applications it is not sufficient to show that a problem instance is satisfiable but an actual solution is also sought for. Finding the solution to a given consistent STP* problem instance can be done by constructing the equivalent STP$^{\neq}$ problem instance implied by the algorithm STP-1 and using for instance the approach found in (Gerevini & Cristani 1997) to find the solution to the STP$^{\neq}$ problem instance. The conversion to STP$^{\neq}$ is done by converting each disjunctive clause $c = x_1 + r_1 \leq y_1 \vee x_2 + r_2 \neq y_2 \vee \cdots \vee x_n + r_n \neq y_n$ that is removed at step 12 of the algorithm to the STP$^{\neq}$ constraints $x_1 + r_1 \leq y_1$ and for each clause remaining in C when the algorithm accepts convert it to the STP$^{\neq}$ constraint $x_i + r_i \neq y_i$ where $i > 2$ and $M[x_i][y_i] \neq r_i \vee M[y_i][x_i] \neq r_i$.

Expressing Interval and Rectangle Algebra Problems in STP*

In this section we discuss how interval and rectangle algebra problems can be expressed in terms of STP* problems. We demonstrate that the full set of preconvex interval algebra relations (ie. the ORD-Horn algebra) can be expressed using STP* and that the expressibility of STP* subsumes that of the preconvex fragment of the augmented interval algebra (Condotta 2000).

The interval algebra was introduced by Allen (1983) and is a relational algebra consisting of 13 atomic relations and their disjunctions. The relations of the interval algebra operate over intervals in a real valued domain are the following: *before (b), after (a), meets (m), met-by (mi), overlaps (o), overlapped-by (oi), during (d), includes (di), starts (s), started-by (si), finishes (f), finished-by (fi)* and *equals (eq)*. An interval algebra network is a set of variables and a set of binary constraints over the variables where each constraint is a disjunction of the atomic relations.

In the rectangle algebra the variables represents rectangles and the constraints disjunctions of tuples of interval constraints over the rectangles projected onto two orthogonal axes. For instance, the constraint $x(b, m)y$ signifies that the rectangle x should come strictly before y on the first axis and should meet y along the second axis.

In order to be able to compare the STP* formalism with the augmented interval algebra we need the definition of it as well. The following definition come from Condotta (2000).

Definition 6 An augmented interval network $\mathcal{M}$ is a pair $\langle \mathcal{N}, \mathcal{S}\rangle$ where $\mathcal{N} = \langle V, C\rangle$ is an interval network which represents the qualitative constraints on the intervals in V and $\mathcal{S} = \langle Points(V), C'\rangle$ is an STP instance representing the quantitative constraints on the distances between the bounds of the intervals in V.

Condotta (2000) has proven that deciding satisfiability for augmented interval networks containing only preconvex relations in its qualitative part can be done in $O(n^5)$ time.

Theorem 7 The satisfiability problem for the augmented interval algebra restricted to problem instances containing only preconvex relations can be reduced to the satisfiability problem for STP* in linear time.

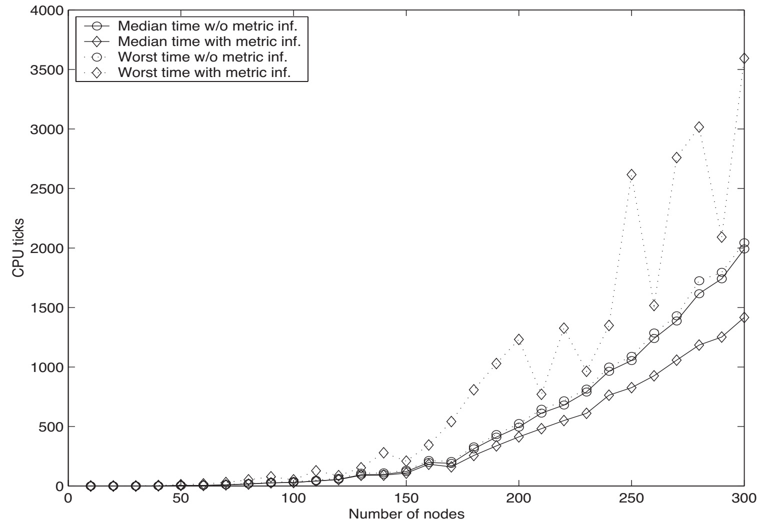

Figure 2: Running time of the STP* solver for converted interval algebra problem instances with and without constraints imposed on the lengths of intervals. For each sample 500 problem instances was generated and solved.

Proof: Let $\mathcal{M} = \langle \mathcal{N}, \mathcal{S} \rangle$ be an augmented interval network such that $\mathcal{N}$ contains only preconvex (ORD-Horn) relations. By definition we know that there exists a constant number of equivalent STP* constraints for each preconvex relation. We construct an STP* problem instance by introducing two STP* variables x^+ and x^- and the constraint $x^- \leq x^+$ for each interval x in $\mathcal{N}$ and translate each qualitative $\mathcal{IA}$ constraint in $\mathcal{N}$ into a constant number of STP* constraints. By also including the STP constraints $\mathcal{S}$ we have an an STP* problem instance which is satisfiable iff $\mathcal{M}$ is satisfiable. □

Furthermore, STP* is able to express constraints which cannot be expressed with only the preconvex relations of the interval algebra. For instance, let $x \ R \ y$ be an interval algebra constraint with a preconvex relation R and C the corresponding set of STP* constraints. If we let C' be C with the term $z^+ \neq w^+$ added to each clause and C'' be C with the term $z^- \neq w^-$ added we can express for instance the non binary constraints $x \ R \ y \vee z \ neq \ w$ as $C' \cup C''$ and $x \ R \ y \vee z \neg (f \vee fi) w$ as C'. We continue by recalling the definition of the augmented rectangle algebra as given by Condotta (2000) and will see that STP* manages to express some of the constraints in this algebra too.

Definition 8 An augmented rectangle network $\mathcal{M}$ is a triple $\langle \mathcal{N}, \mathcal{S}_1, \mathcal{S}_2 \rangle$ where $\mathcal{N} = \langle V, C \rangle$ is a rectangle network which represents the qualitative constraints on the rectangles in V. $\mathcal{S}_1 = \langle Points(V'), C' \rangle$ and $\mathcal{S}_2 = \langle Points(V''), C'' \rangle$ are two STPs representing the quantitative constraints on the distances between the bounds of the interval of V' and V'' respectively, where V', V'' represents the set of the projections of the rectangles in V onto the first and second axis respectively.

In this case, we represent each rectangle with four STP* variables representing the endpoints of the rectangles projected onto each axis. For instance, the rectangle algebra constraint $x(b, mi)y$ representing the fact that x lies left of and meets y from above can be expressed by the STP* constraints $x_1^+ \leq y_1^-$, $x_1^+ \neq y_1^-$, $x_2^- \leq y_2^+$ and $y_2^+ \leq x_2^-$. Obviously, since deciding satisfiability for the full rectangle algebra is an NP-complete problem STP* cannot express all the relations in the full rectangle algebra. Note however that there exists large tractable fragments of the rectangle algebra such as the set of strongly preconvex rectangle algebra relations (Balbiani, Condotta, & Fariñas del Cerro 1999). The full extent of which rectangle algebra relations that can be expressed in terms of STP* constraints remains an open question. Note that all the basic relations in the rectangle algebra can be expressed using STP* constraints.

Evaluating STP*

As was shown in the previous section the expressibility of STP* subsumes that of the preconvex fragment of the augmented interval algebra while still retaining a fairly simple algorithm for deciding satisfiability. In this section we investigate the actual runtime properties of this algorithm. We note that algorithm STP-1 is both simple to implement and gives a practical and efficient method for reasoning about points and intervals with metric information. Although the theoretical time complexity of the algorithm is $O(|C|(|V|^3+|C|t))$ for problem instances $\langle V, C \rangle$ with maximum clause size t (for most applications t is a fixed constant and can be disregarded) it runs in almost cubic time with regard to the number of variables in practice.

In order to test the running time of the algorithm we have first generated random sets of problem instances (without metric information) of the interval algebra. We choose a random graph structure for these problems and with a constant factor chosen from the phase transition region (Cheeseman, Kanefsky, & Taylor 1991) as the average constraintedness. We tested the algorithm on these data and noted that it ran in approximately $O(n^{3.35})$ time for this specific problem ensemble. A reasonable guess considering the behavior of the algorithm would be that adding metric information (in this

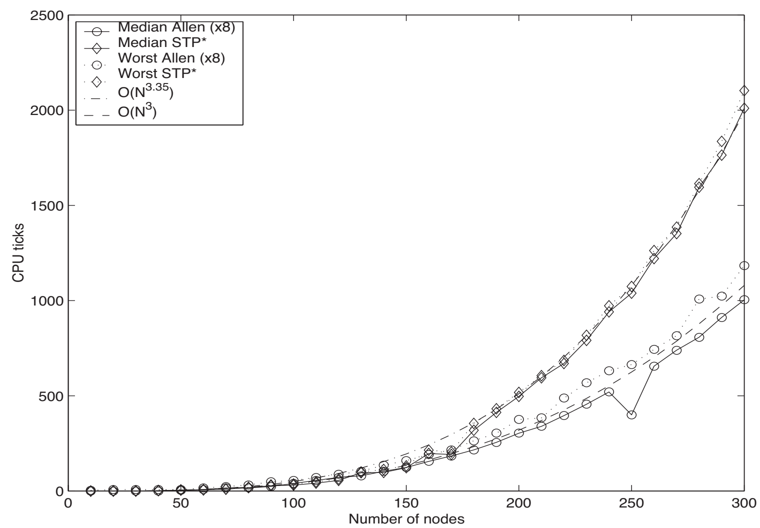

Figure 3: Measured running times for the solvers varying number of nodes in problem instances. For each sample 100 problem instances was generated and solved. Note that the cpu time for the Allen algebra based solver has been exaggerated a factor of eight to better illustrate the asymptotical behavior of the solvers.

case, constraints on the lengths of intervals) will not affect the practical time complexity of the algorithm considerably. As can be seen in Figure 2 adding metric information to our problem instances gives a slight performance increase in the median case, since more problem instances can be found unsatisfiable at an earlier stage, and a small constant performance decrease in the worst case for each problem instance ensemble.

We continue by comparing the running time of this algorithm by that of another solver working directly on the intervals of a problem instance that was used for evaluating the efficiency of using the ORD-Horn fragment (Nebel 1997). For this we generate a number of instances having only qualitative preconvex interval constraints from Allen's interval algebra and solve these instances using both an efficient solver (Nebel 1997) deciding path consistency on the interval constraints directly and using our solver. The conversion of the interval algebra constraints is done by a simple translation procedure containing a case for each possible interval relation, the cost of translating the constraints is thus linear with the number of interval constraints which is quadric with respect to the number of variables. In the graphs we have omitted the cost of translating the constraints since it is assumed that in a real world application the translation should either be done in the modeling of the problem or with a more efficient in-memory approach.

As can be seen the cost of using our solver is relatively low. Since the size of problem instances are doubled when working on point relations, using STP* takes a constant factor of eight times longer. In order to easier compare the asymptotical behavior of the two different approaches this constant factor has been removed in the figures but it is important to remember this extra constant cost induced by the extended expressibility of STP* compared to the interval algebra when solving real world problems. Furthermore, the addition of metric constraints gives a higher time complexity, but in practice not as high as the theoretical $O(n^5)$. Overall, the cost of allowing metric information in problem instances seems to be small and is mostly dominated by the constant factor eight for problem instances of reasonable size. The running times for the two different approaches using the same problem instances can be found in Figure 3. It would here also have been interesting with a comparison of the STP* solver and a solver for Horn DLRs. However, as far as we know there exists no publicly available implementation of such a solver.

Of further interest is also to study the two different approaches with respect to the probability of satisfiability for the problem instances. This is done in Figure 4 by varying the average degree of problem instances. Again, the running times of the approach based on Allen's interval algebra has been scaled by a factor eight to ease comparisons. As we can see the effect of varying the constraintedness of the problem instances is much larger for the $\mathcal{IA}$ approach than for STP* where the running time has very little correlation with the constraintedness of problem instances. This can perhaps best be explained by noting that information is lost in the conversion from interval algebra problems to the point algebra approach used in STP*.

Concluding Remarks

In this paper we have introduced a new extension STP* of the Simple Temporal Problem (STP) which allows disjunctions of disequalities in the constraints and demonstrate how this algebra can be used efficiently for temporal reasoning with metric time. We have given a simple and efficient algorithm for deciding satisfiability of STP* and shown how a solution to a given STP* instance can be found by means of a reduction to STP$^{\neq}$.

The expressibility of STP* extends that of the set of preconvex relations for the augmented interval algebra, which is the only tractable set of relations for this algebra which con-

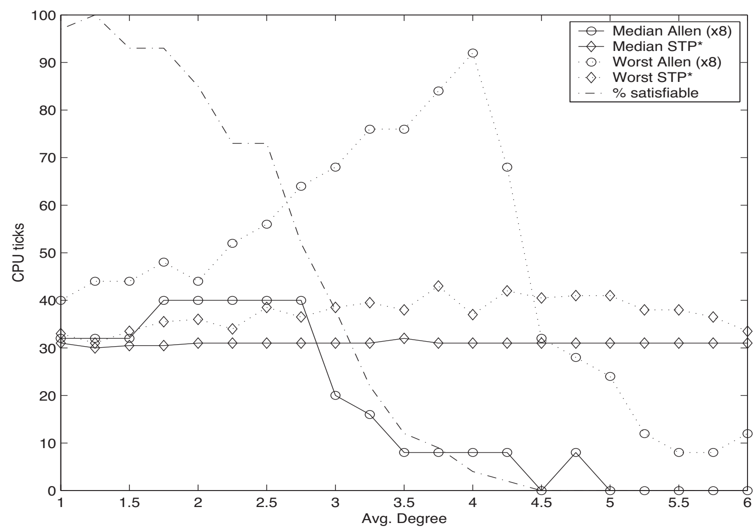

Figure 4: Measured running times for the solvers using 100 problem instances of 100 nodes each and varying the average degree. Note that the cpu time for the Allen algebra based solver has been exaggerated a factor of eight to simplify comparisons.

tains all the basic relations. Furthermore, this algebra can be used not only for temporal reasoning but can also handle for instance fractions of the rectangle algebra.

For future work, it would be interesting to enhance the algorithm so that intervals involving only purely qualitative constraints would be handled as they are, without being converted into two STP* points. This would remove some of the efficiency loss induced by having twice as large problem instances for converted $\mathcal{IA}$ instances. Another open question is that of which rectangle algebra relations can be expressed in terms of STP* constraints. Is it possible to express the full set of strongly-preconvex $\mathcal{RA}$ relations using only the point constraints allowed in STP*?

References

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

Balbiani, P.; Condotta, J.-F.; and Fariñas del Cerro, L. 1999. A new tractable subclass of the rectangle algebra. In *Proceedings of the 16th International Joint Conference on Artificial Intelligence (IJCAI-99)*, 442–447. Stockholm, Sweden: Morgan Kaufmann Publishers.

Cheeseman, P.; Kanefsky, B.; and Taylor, W. M. 1991. Where the really hard problems are. In *Proceedings of the 12th International Joint Conference on Artificial INtelligence (IJCAI-91)*, 331–337. Morgan Kaufmann.

Cohen, D.; Jeavons, P.; Jonsson, P.; and Koubarakis, M. 2000. Building tractable disjunctive constraints. *Journal of the ACM* 47(5):826–853.

Condotta, J.-F. 2000. The augmented interval and rectangle networks. In *Proceedings of the Seventh International Conference on Principles of Knowledge Representation and Reasoning (KR2000)*, 571–579. Trento, Italy: Morgan Kaufmann Publishers.

Cormen, T. H.; Leiserson, C. E.; and Rivest, R. L. 1997. *Introduction to Algorithms*. MIT press.

Dechter, R.; Meiri, I.; and Pearl, J. 1991. Temporal constraint networks. *Artificial Intelligence* 49(1-3):61–95.

Gerevini, A., and Cristani, M. 1997. On finding a solution in temporal constraint satisfaction problems. In *Proceedings of the Fifteenth International Joint Conference on Artificial Intelligence (IJCAI 97)*, 1460–1465.

Güsgen, H. 1989. Spatial reasoning based on allen's temporal logic. Technical Report ICSI TR89-049, International Computer Science Institute.

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

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

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

Nebel, B. 1997. Solving hard qualitative temporal reasoning problems: Evaluating the efficiency of using the ord-horn class. *Constraints* 1(3):175–190. Archive of C-programs: **ftp://ftp.informatik.uni-freiburg.de/documents/papers/ki/allen-csp-solving.programs.tar.gz**.

Papadimitriou, C. H., and Steiglitz, K. 1982. *Combinatorial Optimization: Algorithms and Complexity*. New Jersey: Prentice-Hall.

Pujari, A. K., and Sattar, A. 1999. A new framework for reasoning about points, intervals and durations. In *Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence (IJCAI 99)*, 1259–1267.

Vaidya, P. M. 1987. An algorithm for linear programming which requires $O(((m+n)n^2+(m+n)^{1.5}n)L)$ arithmetic operations. In *Proceedings of the Nineteenth Annual ACM Symposium on Theory of Computing*, 29–38.