To link to this article : DOI:10.1007/978-3-319-10428-7_69
URL : http://dx.doi.org/10.1007/978-3-319-10428-7_69

To cite this version : Cooper, Martin and Maris, Frédéric and Régnier, Pierre *Monotone Temporal Planning : tractability, extensions and applications (Extended abstract).* (2014) In: 20th International Conference on Principles and Practice of Constraint Programming (CP), 8 September 2014 - 12 September 2014 (Lyon, France).

Monotone Temporal Planning: Tractability, Extensions and Applications*
(Extended Abstract)

Martin C. Cooper, Frédéric Maris, and Pierre Régnier

IRIT, University of Toulouse III, 31062 Toulouse, France
{cooper,maris,regnier}@irit.fr

Abstract. We describe a polynomially-solvable class of temporal planning problems. Polynomiality follows from two assumptions. Firstly, by supposing that each fluent (fact) can be established by at most one action, we can quickly determine which actions are necessary in any plan. Secondly, the monotonicity of fluents allows us to express planning as an instance of $STP^{\neq}$ (Simple Temporal Problem with difference constraints). This class includes temporally-expressive problems requiring the concurrent execution of actions, with potential applications in the chemical, pharmaceutical and construction industries. Any (temporal) planning problem has a monotone relaxation, which can lead to the polynomial-time detection of its unsolvability in certain cases. Indeed our relaxation is orthogonal to the relaxation based on ignoring deletes used in classical planning since it preserves deletes and can also exploit temporal information.

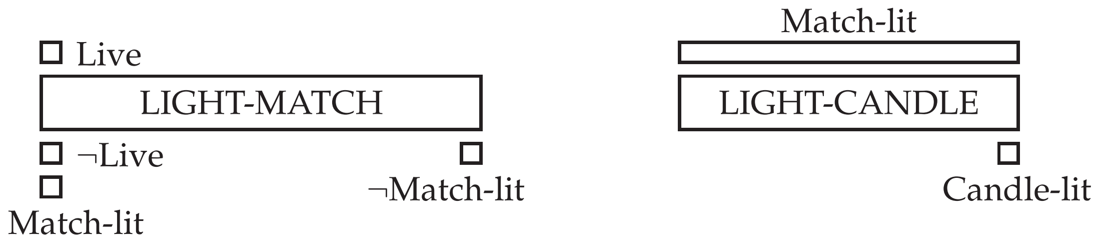
1 Temporal Planning

Temporal planning is an important extension of classical planning in which actions are durative and may overlap. Classical propositional planning is already PSPACE-Complete [1], and temporal planning is EXPSPACE-complete [8]. An important aspect of temporal planning is that, unlike classical planning, it permits us to model so-called temporally-expressive problems in which the execution of two or more actions in parallel is essential in order to solve the problem [5]. We define the first polytime-solvable class of temporal planning. This class includes temporally-expressive problems. It also leads to a novel relaxation of arbitrary temporal planning problems which provides a polytime sufficient condition for the detection of certain properties of actions, fluents and instances. Preliminary (and weaker) versions of this tractable class and temporal relaxation appeared in conference proceedings [2,3] before being improved in the journal paper [4] corresponding to this extended abstract.

A *fluent* is an atomic proposition (such as `door-open`). Changes to the value of a fluent are instantaneous, but conditions on the value of a fluent may be

* This is a summary of the paper Cooper M.C., Maris F., Régnier P., Monotone Temporal Planning: Tractability, Extensions and Applications, *JAIR* 50, 447-485, 2014. This research is supported by ANR Project ANR-10-BLAN-0210.

imposed over an interval. An *action* $a = \langle \mathrm{Cond}(a), \mathrm{Add}(a), \mathrm{Del}(a), \mathrm{Constr}(a) \rangle$ consists of a set Cond(a) of fluents which are required to be true for a to be executed, a set Add(a) of fluents which are established by a, a set Del(a) of fluents which are destroyed by a, and a set Constr(a) of interval constraints between the relative times of events which occur during the execution of a. An *event* corresponds to one of four possibilities: the establishment or destruction of a fluent by an action a, or the beginning or end of an interval over which a fluent is required by an action a. We represent an action by a rectangle whose length corresponds to its duration. Conditions are written above an action, and effects (adds or deletes) below. For example, consider the two actions shown below: LIGHT-MATCH and LIGHT-CANDLE. The action LIGHT-MATCH requires that the match be live, in order to light it. The match remains lit until it is blown out at the end of the action. A constraint in Constr(LIGHT-MATCH) imposes that the duration of the action is at most 10 seconds (at which moment the whole match has burnt). The second action LIGHT-CANDLE requires that the match be lit during two seconds for the candle to be lit.



A *temporal planning problem* $\langle I, A, G \rangle$ consists of a set of actions A, an initial state I and a goal G, where I and G are sets of fluents. In a *positive* problem all fluents in G and Cond(a) (for all actions a) are positive. A *temporal plan* P for the problem $\langle I, A, G \rangle$ is a mapping τ from the events in a set of instances of actions from A to the time dimension such that all conditions of actions are true when required, all goal fluents $g \in G$ are true at the end of the execution of P and the constraints Constr(a) of each action a are satisfied (together with a technical condition ensuring that P is robust under infinitesimal shifts in the starting times of actions). Thus a temporal plan does not schedule its action-instances directly but schedules all the events in its action-instances. A plan is *minimal* if removing any non-empty subset of action-instances produces an invalid plan. For an initial state $I = \{\text{live}, \neg\text{Match-lit}\}$ and a set of goals $G = \{\text{Candle-lit}\}$, it is clear that all minimal temporal plans for our example problem involve executing the two actions in parallel with the start (respectively, end) of LIGHT-MATCH being strictly before (after) the start (end) of LIGHT-CANDLE.

2 Monotonicity and Establisher-Uniqueness Imply Tractability

A set of actions A is *establisher-unique (EU)* if no fluent can be established by two distinct actions of A.

A fluent f is *−monotone** (relative to a positive temporal planning problem $\langle I, A, G\rangle$) if, after being destroyed f is never re-established in any minimal temporal plan for $\langle I, A, G\rangle$. Similarly, a fluent f is *+monotone** if, after having been established f is never destroyed in any minimal temporal plan. A fluent is monotone* if it is either $+$ or $-$monotone*.

An action $a \in A$ is *unitary* for a temporal planning problem $\langle I, A, G\rangle$ if each minimal temporal plan for the $\langle I, A, G\rangle$ contains at most one instance of a. An *action landmark* is an action which occurs in each temporal plan [7].

In our example problem, both actions are clearly essential and hence landmarks. There is only one match available, which means that LIGHT-MATCH can be executed at most once (and is hence unitary). This means that the fluent Match-lit is $-$monotone* since it cannot be established after being destroyed. This same fluent Match-lit is not $+$monotone* since it is destroyed after being established. If $\nexists a_i, a_j \in A$ such that $f \in \mathrm{Add}(a_i) \cap \mathrm{Del}(a_j)$, then f is both $+$monotone* and $-$monotone*. This is the case for $f =$ Candle-lit in our example. In certain IPC benchmark domains (**parcprinter**, **crewplanning**, **tms**), we found that many fluents were monotone* (respectively, 100%, 95% and 50% of those fluents that are either goals or liable to be established in minimal plans).

The following theorem follows from a reduction to STP$^{\neq}$ [6]. The constraints created by this reduction are given in Section 3. The proof of this and all other results are given in the journal version [4] of this extended abstract.

Theorem 1. *The class of positive temporal planning problems $\langle I, A, G\rangle$ in which A is establisher-unique, all fluents are monotone* and all fluents in I are $-$monotone* can be solved in $O(n^3)$ time and $O(n^2)$ space, where n is the total number of events in the actions in A. Indeed, we can even find a temporal plan with the minimum number of action-instances or of minimal cost in the same complexity. Furthermore, if all actions in A are rigid (i.e. intervals between different events in the action are fixed) then the problem of finding a plan with minimum makespan is polytime approximable.*

3 Temporal Relaxation

Relaxation is ubiquitous in Artificial Intelligence. A valid relaxation of an instance I has a solution if I has a solution. Hence when the relaxation has no solution, this implies the unsolvability of the original instance I. A tractable relaxation can be built and solved in polynomial time. Our tractable class of EU monotone planning allows us to define a relaxation TR (temporal Relaxation) which is an alternative to the traditional relaxation of propositional non-temporal planning problems consisting of simply ignoring deletes. In fact, TR is a solution procedure for the class described in Theorem 1 (see [4] for a proof).

We use the notation $a \to f$ (resp., $a \to \neg f$) to denote the event that action a establishes (destroys) fluent f, and $f| \to a$ and $f \to |a$, respectively, to denote the beginning and end of the interval over which action a requires condition f. We use the notation $\tau_{\mathrm{first}}(e)$ (respectively, $\tau_{\mathrm{last}}(e)$) to represent the time in a plan at which an event e occurs first (resp., last).

By applying the following simple rule until convergence we can transform (in polynomial time) any temporal planning problem into a relaxed version which is EU: if a fluent f is established by two distinct actions, then delete f from the goal G and from $\mathrm{Cond}(a)$ for all actions a. From now on we assume the temporal planning problem is EU. We denote by A^{LM} the set of action landmarks that have been detected. Establisher-uniqueness implies that we can easily identify many such actions. The constraints of TR are as follows:

intrinsic constraints: $\forall a \in A^{\mathrm{LM}}$, for all events e of a, $\tau_{\mathrm{first}}(e) \leq \tau_{\mathrm{last}}(e)$.

inherent constraints: $\forall a \in A^{\mathrm{LM}}$, τ_{first} and τ_{last} both satisfy the interval constraints in $\mathrm{Constr}(a)$.

contradictory-effects constraints: no fluent is simultaneously established and destroyed by two actions.

$-$**authorisation constraints:** For each positive fluent f which is known to be $-$monotone*, $\forall a_i, a_j \in A^{\mathrm{LM}}$, if $f \in \mathrm{Del}(a_j) \cap \mathrm{Cond}(a_i)$, then $\tau_{\mathrm{last}}(f \to\mid a_i) < \tau_{\mathrm{first}}(a_j \to \neg f)$. (If $i = j$ then the inequality is not strict [4]).

$+$**authorisation constraints:** For each positive fluent f which is known to be $+$monotone*, $\forall a_i, a_j \in A^{\mathrm{LM}}$, if $f \in \mathrm{Del}(a_j) \cap \mathrm{Add}(a_i)$, then $\tau_{\mathrm{last}}(a_j \to \neg f) < \tau_{\mathrm{first}}(a_i \to f)$.

causality constraints: For each positive fluent f, $\forall a_i, a_j \in A^{\mathrm{LM}}$, if $f \in (\mathrm{Cond}(a_j) \cap \mathrm{Add}(a_i)) \setminus I$ then $\tau_{\mathrm{first}}(a_i \to f) < \tau_{\mathrm{first}}(f \mid\to a_j)$. (If $i = j$ then the inequality is not strict).

goal constraints: $\mathrm{Cond}(A^{\mathrm{LM}}) \subseteq I \cup \mathrm{Add}(A)$, $G \subseteq (I \setminus \mathrm{Del}(A^{\mathrm{LM}})) \cup \mathrm{Add}(A)$, and for each $g \in G$, $\forall a_i, a_j \in A^{\mathrm{LM}}$, if $g \in \mathrm{Del}(a_j) \cap \mathrm{Add}(a_i)$, then $\tau_{\mathrm{last}}(a_j \to \neg g) < \tau_{\mathrm{last}}(a_i \to g)$.

unitary constraint: For each action a which is known to be unitary (see [4] for rules for the polytime detection of unitary actions), for all events e in a, $\tau_{\mathrm{first}}(e) = \tau_{\mathrm{last}}(e)$.

Theorem 2. *A temporal planning problem in the tractable class described in Theorem 1 has a solution if and only if TR has a solution.*

We can use TR to detect certain properties of actions, fluents and problems.

Lemma 1. *If the temporal relaxation TR(I, A, G) of a positive temporal planning problem $\langle I, A, G \rangle$ has no solution, then $\langle I, A, G \rangle$ has no solution. If TR($I, A \setminus \{a\}, G$) has no solution, then a is a landmark action in $\langle I, A, G \rangle$.*

The detection of monotonicity* is theoretically as difficult as temporal planning, since it is EXPSPACE-complete [4]. However, TR together with extra constraints provides a powerful polytime method for detecting monotonicity*.

Lemma 2. *If $\forall a, b \in A$ s.t. $f \in \mathrm{Add}(a) \cap \mathrm{Del}(b)$, TR together with the constraint $\tau_{\mathrm{first}}(a \to f) < \tau_{\mathrm{last}}(b \to \neg f)$ is inconsistent, then f is $+$monotone*. If $\forall a, b \in A$ s.t. $f \in \mathrm{Add}(a) \cap \mathrm{Del}(b)$, TR together with the constraint $\tau_{\mathrm{first}}(b \to \neg f) < \tau_{\mathrm{last}}(a \to f)$ is inconsistent, then f is $-$monotone*.*

The class of temporal planning problems described in Theorem 1 which also have the property that all fluents can be detected as monotone* by Lemma 2 constitutes a tractable class that can be detected and solved in polynomial time.

Further research is required to determine if interesting tractable classes can be defined without the restrictive assumption of establisher-uniqueness.

References

1. Bylander, T.: The Computational Complexity of Propositional STRIPS Planning. Artificial Intelligence 69(1-2), 165–204 (1994)
2. Cooper, M.C., Maris, F., Régnier, P.: Tractable monotone temporal planning. In: Proceedings ICAPS 2012, pp. 20–28 (2012)
3. Cooper, M.C., Maris, F., Régnier, P.: Relaxation of Temporal Planning Problems. In: International Symposium on Temporal Representation and Reasoning (TIME), pp. 37–44 (2013)
4. Cooper, M.C., Maris, F., Régnier, P.: Monotone Temporal Planning: Tractability, Extensions and Applications. JAIR 50, 447–485 (2014), `http://www.irit.fr/publis/ADRIA/MonotoneJAIR.pdf`
5. Cushing, W., Kambhampati, S., Mausam, W.D.S.: When is Temporal Planning Really Temporal? In: Proc. International Joint Conference on Artificial Intelligence (IJCAI 2007), pp. 1852–1859 (2007)
6. Gerevini, A., Cristani, M.: On Finding a Solution in Temporal Constraint Satisfaction Problems. In: Proc. International Joint Conference on Artificial Intelligence (IJCAI 1997), pp. 1460–1465 (1997)
7. Karpas, E., Domshlak, C.: Cost-optimal planning with landmarks. In: Proc. International Joint Conference on Artificial Intelligence (IJCAI 2009), pp. 1728–1733 (2009)
8. Rintanen, J.: Complexity of Concurrent Temporal Planning. In: Proc. 17th International Conference on Automated Planning and Scheduling (ICAPS 2007), pp. 280–287 (2007)