Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence

Planning with Numeric Timed Initial Fluents

Chiara Piacentini, Maria Fox, Derek Long
Informatics Department
King's College London
London, UK
email: name.surname@kcl.ac.uk

Abstract

Numeric Timed Initial Fluents represent a new feature in PDDL that extends the concept of Timed Initial Literals to numeric fluents. They are particularly useful to model independent functions that change through time and influence the actions to be applied. Although they are very useful to model real world problems, they are not systematically defined in the family of PDDL languages and they are not implemented in any generic PDDL planner, except for POPF2 and UPMurphi.

In this paper we present an extension of the planner POPF2 (POPF-TIF) to handle problems with numeric Timed Initial Fluents. We propose and evaluate two contributions: the first is based on improvements of the heuristic evaluation, while the second considers alternative search algorithms based on a mixture of Enforced Hill Climbing and Best First Search.

1 Introduction

Timed Initial Literals (TILs) were first introduced in PDDL during the IPC-4 to model deterministic unconditional exogenous events that become true or false at time points. Numeric Timed Initial Fluents (TIFs) are a natural extension of TILs that express changes of numeric fluents, allowing the addition or the deletion of resources independently of actions in the plan.

We only consider exogenous events that assign values to fluents and do not allow relative changes by increase or decrease effects. Such effects can be reformulated using only TIF assignment by decomposing fluents into two parts, one modified by actions and the other by TIFs.

For example, given a domain fluent `(x)`:

```
(at 1 (= (x) 1))
(at 2 (= (x) 4))
```

In real world problems, TIFs are often used to describe discretised functions of time that are independent of the actions chosen by the planner. In particular, there is a class of "real world" problems, such as the Unit Commitment Problem (UCP) (Campion et al. 2013) and the Electrical Balancing Problem (EB) (Piacentini et al. 2013), characterised by the presence of background functions (in these cases the electrical demand profiles), against which the planner needs to actively react to maintain some numeric constraints through an extended period of time.

Despite the practical value of TIFs, currently no IPC benchmark domain with TIFs is available and only the PDDL planners POPF2 (Coles et al. 2010) and UPMurphi (Della Penna et al. 2009) are capable of dealing with them directly, treating them just as an extension of TILs.

2 Planning with Exogenous Events

The effect of exogenous events (both TILs and TIFs) is generally to delay resources, remove resources or violate active conditions. The latter case occurs when the problem has actions with `over all` conditions (conditions that must hold during the entire duration of an action) or trajectory constraints. The major difference between TILs and TIFs is due to the presence of active constraints. In the case of a propositional domain, an active constraint involves only a proposition that is required to be *false* or *true* throughout the duration of the action. If a TIL modifies that proposition, it makes the precondition of that action *true* (or *false*) and we can deduce that the action cannot intersect the time-point of the TIL.

An interesting case arises when the condition contains some expressions including a fluent changed by TIFs and some other fluent controlled by actions. In this case, if the condition would be violated because of the application of a TIF, the planner can choose to apply an action to balance the value of the other fluent expression. When the action with the `over all` constraint cannot be postponed or timed to avoid the critical TIFs, we can see that applying the correct balancing action is essential to find the solution. For this, when we add TIFs in a planning problem, the planner needs an extension that is beyond the simple extension of TILs to numeric fluents.

3 Implementation of POPF-TIF

We extend POPF2 (Coles et al. 2010) to POPF-TIF by the modification of the heuristic and two new alternative search algorithms.

3.1 Modifying the Heuristic Evaluation

It has been observed (Baier and Botea 2009) that the FF relaxed plan heuristic sometimes oversimplifies the struc-

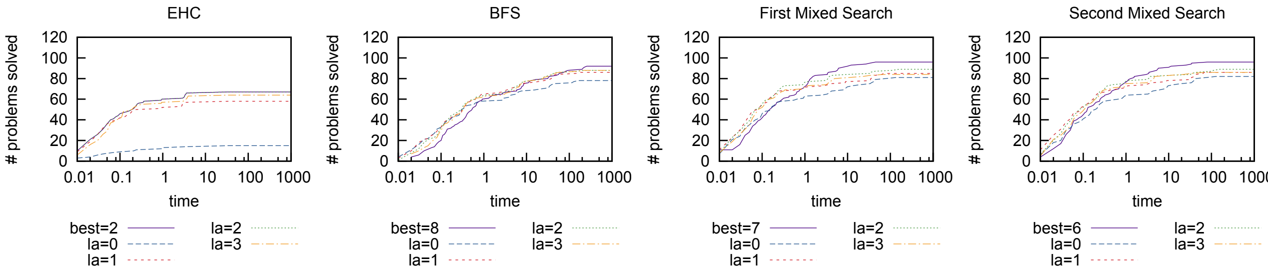

Figure 1: Number of problems solved as function of execution time.

ture of a problem, resulting in an inaccuracy of the heuristic value. A way to mitigate this is to reduce the relaxation to gain more information. In the paper (Piacentini et al. 2013), this idea is applied for TIFs: during the extraction of the relaxed plan, starting from the layer containing the goal, the algorithm proceeds backwards, applying the actions needed to achieve the goals. When a the fact layer that corresponds to the first TIF is met, then the `over all` constraints of *open actions* are checked for violation, because of the TIF, and, in the case of conflict, the violated constraints are added to the goals of the relaxed plan graph.

In our work we extend this *lookahead* mechanism allowing more TIFs to be considered in the heuristic evaluation. Domains involving TIFs often have a rich temporal structure, and one *lookahead* might not be adequate.

3.2 Additional Search Algorithms

We want to take advantage of the local search EHC benefits, but decrease the possibility of failure. For this reason we use an algorithm that allows passage from the EHC into BFS, exploiting the information that EHC already acquired. In addition, we want to be able to switch back to the EHC when a "safe" state is reached.

We propose two different search algorithms based on the mixture of EHC and BFS, that we call S1 and S2 . The difference between them is in how the states are selected from a *backup queue*. In the first case (S1) we resort to the BFS from the last *backup states* of the queue and continue until we find a solution or a *restoring state*. In the latter case we switch back to the EHC. If the BFS fails we can invoke a new BFS from the latest state in the *backup queue*. In the second algorithm (S2), when the BFS is activated, it starts with the entire *backup queue*, so it begins with some states explored. Also in this case, when a *restoring state* is found we switch back to the EHC.

4 Experimental Evaluation

We now present results of the planner POPF-TIF with the following domains (30 problems each domain):

- **Unit Commitment Problem (UCP)**;
- **Rover domain**: a variant of the numeric rover domain in which the rover is equipped with a solar panel that gives energy to the rover according to the exposure to the sun.
- **Temperature domain**: a PDDL version of the problem presented in (Ono, Graybill, and Williams 2012), where the objective is to maintain the temperature of a smart home within the user's preferences. The temperature is subject to weather conditions, but can be regulated by means of electrochromic windows, that can be tinted.
- **Skier domain**: a skier descends along a track at constant velocity, but the track is not always straight, so the problem for the skier is to decide when to turn left or right to arrive to the goal without going off track.

In our evaluation we compare EHC, BFS, S1 and S2. For each search strategy we use heuristic evaluations with a different number of *lookaheads*, from 0 to 10, where 0 indicates the standard heuristic evaluation. All tests used a 3.4 GHz Intel Core i7-2600 machine, limited to 30 minutes and 4 GB of memory. In Fig. 1 we plot the number of problems solved as a function of execution time for all the domains. The results indicate that adding more lookaheads increases the coverage of all the four search strategies. The algorithm most affected by the lookahead is EHC, which benefits from the change in the heuristic mainly because the relaxed plan graph includes a more appropriate set of helpful actions. Overall the algorithm with more coverage is the first of the two mixed search algorithms.

References

Baier, J. A., and Botea, A. 2009. Improving Planning Performance Using Low-Conflict Relaxed Plans. *In Proc. Int. Conf. Automated Planning and Scheduling (ICAPS).*

Campion, J.; Dent, C.; Fox, M.; Long, D.; and Magazzeni, D. 2013. Challenge: Modelling Unit Commitment as a Planning Problem. *In Proc. Int. Conf. Automated Planning and Scheduling (ICAPS).*

Coles, A. J.; Coles, A.; Fox, M.; and Long, D. 2010. Forward-Chaining Partial-Order Planning. *In Proc. Int. Conf. Automated Planning and Scheduling (ICAPS).*

Della Penna, G.; Magazzeni, D.; Mercorio, F.; and Intrigila, B. 2009. UPMurphi: A Tool for Universal Planning on PDDL+ Problems. *In Proc. Int. Conf. Automated Planning and Scheduling (ICAPS).*

Ono, M.; Graybill, W.; and Williams, B. C. 2012. Risk-sensitive plan execution for connected sustainable home. In *BuildSys '12 Proc. of ACM Workshop on Embedded Sensing Systems for Energy-Efficiency in Buildings*. ACM Press.

Piacentini, C.; Alimisis, V.; Fox, M.; and Long, D. 2013. Combining a Temporal Planner with an External Solver for the Power Balancing Problem in an Electricity Network. *In Proc. Int. Conf. Automated Planning and Scheduling (ICAPS).*