Temporal-Numeric Planning with Infinite Domain Action Parameters

Emre Savaş and **Stefan Edelkamp** and **Maria Fox** and **Derek Long** and **Daniele Magazzeni**
Department of Informatics, King's College London, London, UK, WC2B 4BG
firstname.lastname@kcl.ac.uk

Abstract

A limitation of task planning as a search problem is that action parameters are restricted to take their values from a finite collection of objects defined in a problem instance. In real-world, there are parameters that have freedom of taking their values from infinite domains, which we call *control parameters*. Planning with control parameters are deemed to be more than just a modelling choice; but it is a modelling requirement in real-world applications. POPCORN is a partially-ordered forward state space search planner that can reason with multiple *typed* control parameters declared in action schemes. In this paper, we briefly introduce the planning with control parameters paradigm.

1 Introduction

Over the last decade, significant progress has been made in task planning to explore temporal and numerical features of the real world, including the work of (Do and Kambhampati 2003; Coles et al. 2012; Eyerich, Mattmüller, and Röger 2012; Benton, Coles, and Coles 2012; Piotrowski et al. 2016). Interaction between time and metric fluents and temporal coordination problems have been well-addressed by innovative approaches in planning and scheduling. A standardised language, PDDL, has enabled modelling the conceptual models and benchmarking the work of numerous researchers since it was released.

Although PDDL is an expressive modelling language, a significant limitation is imposed on the structure of actions: the parameters of actions are restricted to values from finite (in fact, explicitly enumerated) domains. There is one exception to this, introduced in PDDL2.1 (Fox and Long 2003), which is that durative actions may have durations that are chosen (possibly subject to explicit constraints in the action models) by the planner. A motivation for this limitation is that it ensures that the set of grounded actions is finite and, ignoring duration, the branching factor of action choices at a state is therefore finite. Although the duration parameter can make this choice infinite, very few planners support this possibility, but restrict themselves to durative actions with fixed durations.

In this paper, we briefly introduce planning with control parameters paradigm in a temporal setting. We provide brief details of the effects of control parameters in the forwards search space. We conclude the paper with a set of real-world examples where modelling using control parameters is more than just a modelling choice. This work is described in detail in work of Savaş et al. (2016).

2 A Motivating Case

We start with a very simple example motivating the use planning with control parameters. Suppose we are given two jugs and we want to pour *a bit* of water from the one into the other as illustrated in Figure 1. We know that the water level in the one jug will decrease by the amount the other one will increase, and we also observe obvious side constraints on the pouring actions. However, during modelling we do not yet know the numerical amount of water being poured, as this will be dependent on the partial plan executed so far and has to be optimised for plan improvement.

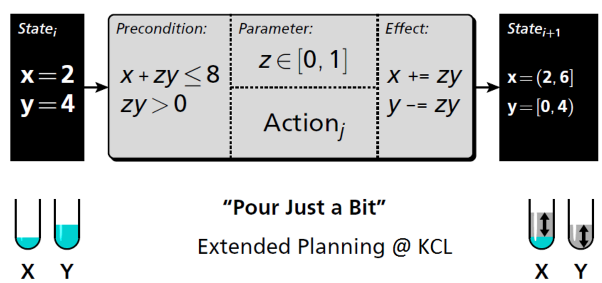

Figure 1: Water pouring action with control parameters.

3 Planning with Control Parameters

Planning with control parameters is a generalisation of the duration parameter of durative actions to other types, allowing actions to take multiple duration-independent infinite domain parameters. We define the temporal and numeric planning problem with control parameters as follows:

Definition 1 *A temporal and numeric planning problem with control parameters is a tuple* $\langle F, \vec{v}, I, A, G, M \rangle$*, where:*

- F *is a finite set of grounded literals,*
- $\vec{v}$ *is a finite set of numeric variables,*

- I is the initial state,
- A is a set of actions. Each action, $a \in A$, is a tuple:
  $a = \langle dur, cont, pre^n_x, pre_x, eff^n_y, eff_y \rangle$
  - *dur* is a set of duration constraints of action a, where each is expressible in the form of $\langle g(\vec{v}, \vec{d^a}, \texttt{?duration}), comp, c \rangle$, where $c \in \mathbb{R}$.
  - pre_x and pre^n_x are the propositional and numeric conditions, respectively, that must hold at a state in which the action starts ($\vdash$), at a state in which the action ends ($\dashv$), or between the start and the end of an action a ($\leftrightarrow$): $x \in \{\vdash, \leftrightarrow, \dashv\}$. The numeric conditions are in the form:
  $$pre^n_x = \langle f(\vec{v}, \vec{d^a}), comp, c \rangle$$
  - eff_y are the propositional start (or end) effects of an action a that are to be added to (or deleted from) the world state. eff^n_y are the numeric effects that acts on variable $v \in \vec{v}$. We denote the decrease effects by eff^-_y and the increase effects by eff^+_y; where $y \in \{\vdash, \dashv\}$. The numeric effect eff^n_y is expressible in the form:
  $$eff^n_y = \langle v, op, g(\vec{v}, \vec{d^a}, \texttt{?duration}) \rangle,$$
- G is the goal described by a set of propositions, $p(G) \subseteq F$, and a set of numeric conditions, $v(G)$, over the state variables,
- M is the metric objective function.

A solution to such a problem is an extended version of the plans described in (Coles et al. 2012):

Definition 2 (Controlled Plan (π)) *is a list of timestamped actions with duration and specified values for control parameters of actions in* π, $\{\langle a, t, dur, \vec{d^a} \rangle, \dots\}$, *where* $t \in \mathbb{R}$ *is a timestamp (the time at which the action* a *is applied),* $dur \in \mathbb{R}$ *is the duration and* $\vec{d^a} \in \mathbb{R}^{m^a}$ *is the vector of values of the control parameters of* a. *The duration constraints of each action in* π *are satisfied, and all* pre^n_x *of actions in* π *hold in states in which the actions are applied. The trajectory of states generated by application of the actions yields a final state that satisfies the goal.*

POPCORN is a forward state space heuristic search planner. It builds a *constraint space* search alongside the forwards search as follows. It collects temporal-numeric constraints that appears in the conditions and effects of actions chosen during forwards search. It uses Simple Temporal Network (STN) to check temporal consistency at a finitely branching state. However, the existence of control parameters creates infinite branching factor in forward state space search. To avoid infinite branching decision, POPCORN planner uses a Linear Programming (LP) solver to check the temporal/numeric consistency at a state. It leaves infinite branching choice to the constraint space, relying on LP-solving. LP solver helps to prune states with infeasible solutions. Figure 2 illustrates the resulting search space.

4 Conclusion

We introduced a way of modelling infinite domain action parameters in expressive temporal-numeric PDDL language. The approach accumulates constraints during planning and uses them to determine the values of these parameters and to prune inconsistent branches from the search. The use of control parameters expands the real-world applicability of task planning.

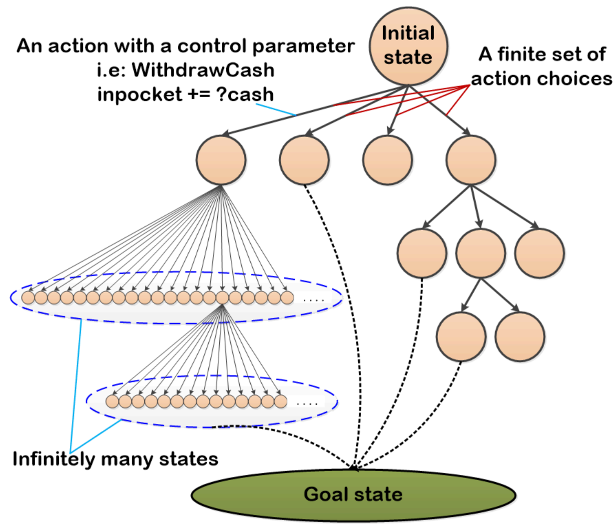

Figure 2: Schematic representation of the search space where there is a control parameter effect. The nodes represent the state reached, and the edges represent the action applied to reach the next state. The graphs in black boxes represent the LP constraint space, which is used to avoid complex branching choice.

References

Benton, J.; Coles, A.; and Coles, A. 2012. Temporal Planning with Preferences and Time-Dependent Continuous Costs. In *Proceedings of the Twenty-First International Conference on Automated Planning and Scheduling, ICAPS*.

Coles, A.; Coles, A.; Fox, M.; and Long, D. 2012. COLIN: Planning with Continuous Linear Numeric Change. *Journal of Artificial Intelligence Research (JAIR)* 1–96.

Do, M. B., and Kambhampati, S. 2003. Sapa: A multi-objective metric temporal planner. *Journal of Artificial Intelligence Research* 20:155–194.

Eyerich, P.; Mattmüller, R.; and Röger, G. 2012. Using the context-enhanced additive heuristic for temporal and numeric planning. *Towards Service Robots for Everyday Environments* 76:49–64.

Fox, M., and Long, D. 2003. PDDL2.1: An Extension to PDDL for Expressing Temporal Planning Domains. *Journal of Artificial Intelligence Research (JAIR)* 20:61–124.

Piotrowski, W.; Fox, M.; Long, D.; Magazzeni, D.; and Mercorio, F. 2016. Heuristic Planning for PDDL+ Domains. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence (IJCAI)*. AAAI Press.

Savaş, E.; Fox, M.; Long, D.; and Magazzeni, D. 2016. Planning Using Actions with Control Parameters. In *Proceedings of the Twenty-Second European Conference on Artificial Intelligence (ECAI 2016)*, 1185–1193. IOS Press.