Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence (AAAI-10)

Temporal Planning for Interacting Durative Actions with Continuous Effects

Serdar Kecici and **Sanem Sariel Talay**
Department of Computer Engineering
Istanbul Technical University, Istanbul, Turkey
{kecici,sariel}@itu.edu.tr

Introduction

One of the most important properties of temporal planners is their capability of integrating planning into scheduling so that time can be considered as an optimization objective. Forward chaining temporal planners provides tight coupling of planning and scheduling. In these planners, a search node is represented by a world state including the applied action directly attached with a world clock which defines starting time of that action. The search proceeds by applying a new action or advancing the world clock which moves the time forward. To avoid infinite branching factor, forward chaining temporal planners can set a world clock only at certain points in time called *decision epochs* for which an *at-end* effect of a previously applied action is scheduled. Since all necessary timestamps in continuous time line are not covered, the resulting plan may be suboptimal for planning problems that include both discrete effects and time-dependent continuous linear effects. Continuous linear effects occur especially when agents share time-dependent critical resources. In these cases, besides discrete and continuous changes, their interactions should also be taken into consideration.

Some of the earlier studies have investigated planning domains including actions with continuous linear effects. TM-LPSAT (Shin & Davis 2005) is an extension to SAT-based planning framework to handle concurrent actions with continuous change. COLIN (Coles *et al.* 2009) is another effective planner which plans with mixed discrete continuous numeric changes. Both of these planners support PDDL+ (Fox & Long 2006) language which provides representation of continuous process effects in planning domains. However, the focus of both systems is to handle increase and decrease in the value of singular variables. Neither of the studies considers interactions among continuous effects of different actions in planning domain.

This paper presents an extension to forward chaining temporal planning to handle continuous linear effects of actions and interactions among them. The main motivation is providing completeness considering these effects.

We propose an action lifting approach in order to ensure completeness in domains with interacting linear continuous effects. VHPOP system (Younes & Simmons 2003) as a partial order causal link planner also applies lifted actions to reduce branching factor of their algorithm. (Cushing *et al.* 2007) proposes lifting states over time in order to provide completeness for domains including actions with *at-end* conditions. However both of them cannot handle continuous effects of actions.

We implemented our algorithm as an extension to the TLPLAN planning system (Bacchus & Ady 2001), a progression temporal planner which handles durative concurrent actions and metric quantities.

Problem Statement

We analyze the path sharing problem to illustrate interaction of continuous linear effects in the planning domain. In addition to discrete changes, this problem presents further relations such as interactions of continuous changes. In a path sharing problem, such interaction appears when agents move on the path simultaneously.

Multiple agents, defined in the planning domain, carry out several tasks in order to reach the goal state of the planning problem. There are several rooms connected with a shared narrow corridor which is a one-lane shared path. Figure 1 shows the problem domain. Agents could move on the path at predefined speed unless they collide with any other agent concurrently moving on the shared path.

Critical resource that agents must share in our domain is the shared path of which availability cannot be represented by a singular predicate, but by several allocations. An allocation is a linear continuous effect of a move action which is formulated by 4 values: speed of the agent, entrance position and exit positions on the path and start-time of the movement. As simultaneous movement of agents on the path is possible, there may be multiple concurrent allocations to represent the availability of the critical resource.

Lifted Actions

Temporal planners do not consider continuous time, but plans with limited set of timestamps called *decision epochs*. This set of *decision epochs* includes the timestamps where world state is modified by *at-end* effect of an action. Although this usage is adequate for domains in-

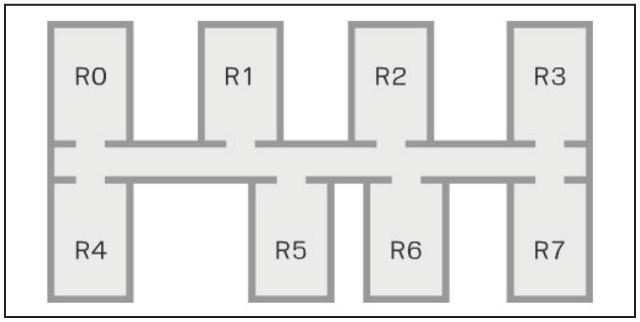

Figure 1 : Domain with a shared path and seven rooms

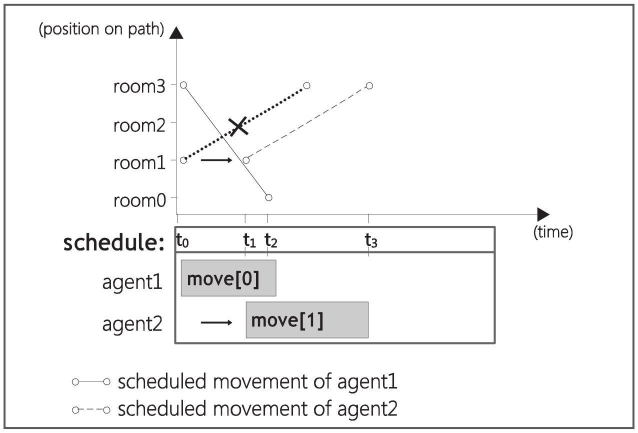

Figure 2 : Solution for a sample problem with two agents

volving actions with ordinary preconditions, it is not sufficient for complex actions which are related with continuous changes on shared critical resources (e.g., shared path) especially when the metric to minimize is the total duration of the plan.

We illustrate this incompleteness with a simple problem instance. Initially $agent_1$ is in $room_3$, and $agent_2$ is in $room_1$. The goal is defined as $agent_1$ in $room_0$ and $agent_2$ in $room_3$. Figure 2 illustrates the solution scenario for this simple problem.

In this problem, after applying the move action of $agent_1$, the search process will fail to match preconditions of move action for $agent_2$ at t_0 because of the collision of requested allocation with the allocation already set by the already scheduled movement of $agent_1$. So *advance-world-clock* action is applied. As classical forward chaining mechanism considers only the set of *decision epochs*, this action will advance the world clock to t_2. Although an allocation for the movement of $agent_2$ starting at t_1 does not collide with any other allocation, it is missed by the planner since t_1 is not a *decision epoch*. Therefore, the planner cannot find the makespan optimal solution. It finds a sub-optimal plan in which the movement of $agent_2$ is applied at t_2 instead of t_1. This is the strategy performed by TLPLAN. To overcome this problem, we extend the planner to broaden its search by applying lifted actions in addition to standard actions. Lifting is decision of an action application without scheduling it to a certain time point. The actual start time of a lifted action is left open until the next decision epoch where the lifted action may be grounded (scheduled) to an intermediate time point.

There are 4 successor generators for the planner while expanding a state *s*.

a_i : i-th action of the domain
$ts(a_i)$: actual start time of action a_i
$p(a_i)$: classical preconditions of a_i
$p_c(a_i)$: conditions related with continuous changes
L : set of lifted actions

- Schedule (apply) an action a_i : $ts(a_i)$= *world-clock* of s
  If $p(a_i)$ is met at s
  and $p_c(a_i)$ is met for world clock of s
- Advance the world clock to the next *decision epoch*
  If set *decision epoch* is not empty
- Lift (apply without scheduling) an action a_i: $L = L \cup \{a_i\}$
  if $p(a_i)$ is met at s
  and $p_c(a_i) \neq \varnothing$
- Ground an action a_i :
  $ts(a_i)$=*earliest-possible-start-time* for a_i
  if $a_i \in L$
  and $p_c(a_i)$ is met for $ts(a_i)$
  and $ts(a_i) \notin$ set of *decision-epochs*

This extended branching scheme reveals complete and optimal solutions for the given scenario. As it is illustrated in figure 2, it is possible for the planner to lift the move action of $agent_2$ at t_0, so that it might be grounded (scheduled) to t_1 after advancing the world clock to t_2.

Conclusion

In this paper, we introduce an extension to forward chaining planning in order to solve planning domains with continuous linear change (e.g., domains with a shared path). Such domains may involve mixed discrete continuous changes and interactions among them. We show that lifting can be adapted to handle continuous change in planning domains so that optimal solutions in terms of makespan minimization are provided. However the drawback of this approach is the increased computational complexity due to the extended branching factor. Special pruning techniques for temporally identical states and use of heuristics for makespan estimation will be investigated in future studies to reduce complexity.

References

Bacchus, F.; Ady, M. 2001. Planning with resources and concurrency: A forward chaining approach. In *Proc. of IJCAI.*

Coles, A. J.; Coles, A. I.; Fox, M.; and Long, D. 2009. Temporal planning in domains with linear processes. In *Proc. of IJCAI.*

Cushing,W., Kambhampati, S.; Mausam M.; and Weld, D. 2007. When is temporal planning really temporal?. In *Proc. of IJCAI.*

Fox, M., and Long, D. 2006. Modelling mixed discrete-continuous domains for planning. *JAIR, 27:235–297.*

Shin, J.-A., and Davis, E. 2005. Processes and continuous change in a SAT-based planner. *Artificial Intelligence* 166(1-2):194–253.

Younes, H.; and Simmons, R. 2003. Vhpop: Versatile heuristic partial order planner. *JAIR* 20:405–430.