Proceedings of the Sixteenth International Symposium on Combinatorial Search (SoCS 2023)

Domain Specific Situated Planning (Student Abstract)

Devin Wild Thomas

University of New Hampshire
dwt@cs.unh.edu

Abstract

Traditionally when planning we assume that we receive the problem instance as input, then formulate a plan, then the clock begins and the agent executes the plan. Sometimes we are forced to consider time passing as we plan, which is known as situated planning. In my dissertation I explore situated planning in three domains: grid based path planning, the orienteering problem and opportunistic science.

Introduction

The first domain I explore situated planning in is grid based path planning, where the environment includes both static and moving obstacles and the agent seeks to minimize its goal achievement time while avoiding collision with any obstacles and a popular algorithm for solving it offline are known as safe interval path planning (SIPP)(Phillips and Likhachev 2011). Second, in the orienteering problem, where the agent is given a graph containing vertices with varying rewards and has a time limit to visit some portion of those nodes and accumulate as much total value as it can. Third, in the problem of opportunistic science, where the agent may have a window of time to amend its current plan to take advantage of a transient measurement opportunity. In the remainder of this abstract I will give an overview of the background of situated planning, and each of the domains covered by the dissertation.

Situated Planning

Russell and Wefald (1991) argue that computations are actions, and the utility of such an action should be derived from its effect on the agent's choice of actions. This utility can be estimated from statistical knowledge of the utility of previous similar computation actions, or heuristic estimates. More recently the problem of situated planning was posed by Cashmore et al. (2018), where the planner takes into account that execution is waiting on planning. This allows the planner to prune partial plans where the planning would likely finish too late to execute. In domain independent planning with absolute deadlines, this showed empirical improvements over a baseline planner that had a set planning time and moved all the deadlines earlier by that same planning time. Shperberg et al. (2019) formalize the 'metareasoning problem' allocating planning effort when actions expire (AE2) where the situated agent must decide which search nodes to expend planning effort on, when they each have an expiration time and expected completion effort in addition to the normal cost. They optimize to maximize the probability that at least one solution is found by the deadline. They develop a formal MDP solution for AE2, and empirically demonstrate a greedy scheme that was near optimal in solution quality, while also fast enough to be used in metareasoning. If planning is slow enough to require taking duration into account, we have the contract search setting, in which the start time functions as a deadline by which planning must complete (Dionne, Thayer, and Ruml 2011). Situated planning is an area of active work, and it remains unclear how much of the sophisticated theoretical work can be applied to concrete problems. This dissertation focuses on three different specific domains where we can concretely test the utility of these metareasoning algorithms.

Three Problem Domains

This first domain is a 2D grid with static and moving obstacles; this setting has been the focus of my work thus far. The others are orienteering and opportunistic science, which I have discussed as potential domains to explore, but have not begun working in yet. I chose to start with grid pathfinding because I (naively) assumed it would be simple to extend to a situated agent. While this has not been the case, our work on situated grid pathfinding has exposed many interesting questions that I have started to address.

Situated Safe Interval Path Planning

Many applications of planning involve planning on a time dependent graph. Often this time-dependence can be represented as safe intervals where a state or action is either safe or unsafe at any moment in time. In SIPP we assume the agent knows the plan starting time. SIPP (the algorithm) solves SIPP (the problem) by noticing that, while there are an infinite set of timed trajectories the agent can decide from, as it can wait for any amount of time anywhere along its path, arriving earlier is always at least as good as being there at any later time, because you could simply wait to reach any later time. SIPP uses this dominance relation to search on the space of safe intervals, rather than on points in space and

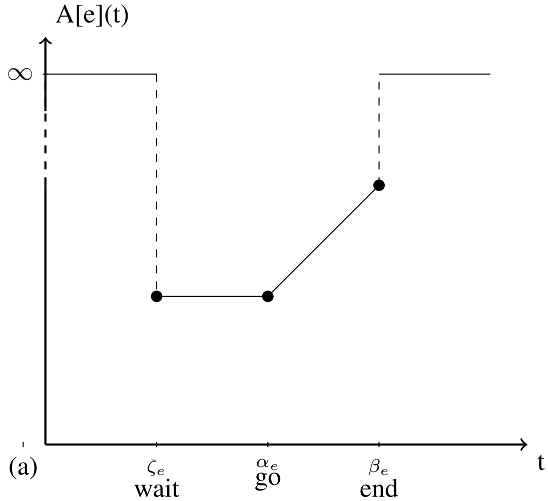

Figure 1: Arrival Time Function

time. This state dominance relation no longer holds in the situated setting. Arriving early means the agent could have spent more time planning prior to committing to the action that was taken, and perhaps that additional planning would have found a better path! To be correct, we would need to be able to track when we expect the agent to arrive as a function of its departure time. With this, we could metareason knowing what opportunity cost we are expecting to pay for additional time spent planning. In fact, a much simpler alteration to the SIPP domain results in the same requirement. If we relax our assumption that we know the starting time of the navigation, this also negates the ability to search with just the earliest arrival time as SIPP does and directly requires us to reason with arrival time functions. We call this any-start-time planning, and are working on an any-start-time SIPP.

Any-start-time Safe Interval Path Planning

While SIPP is able to search by tracking only a single time point for the g value, we adapt concepts from the time dependent shortest path literature (Foschini, Hershberger, and Suri 2014) and search tracking piecewise linear arrival time functions as g, as seen in Figure 1. Our experimental results show that there is negligible runtime overhead associated with tracking arrival time functions instead of simply the earliest arrival time. We have developed an 'augmented' SIPP that returns the whole family of related paths captured by a single arrival time function instead of the single path returned by SIPP. This is a useful addition to SIPP, as it tells the agent when the path it has planned expires (β in Figure 1) as well as if there is any slack time it could wait without delaying its arrival at the destination (α in Figure 1). Additionally we have shown that there are instances where paths are invalidated so quickly that if SIPP waits until it learns the start time, it is unable to generate a plan before that plan becomes invalid! We have also developed a partial-expansion-based solution to the any-start-time problem that progressively finds optimal plans for later and later start times. In our future work we intend to adapt the arrival time functions to allow heuristic learning, to aid metareasoning in situated planning, and work on a situated SIPP.

This initial grid problem setting provides an arena to test how methods from real-time planning can be adapted to the situated setting. Our initial results suggest that with appropriate choices of state space representation and learning algorithm, a situated agent can perform very well in these instances. We intend to construct instances of our own: for example, instances of the game Frogger/Crossy Road would also be suitable to benchmark situated SIPP agents.

Orienteering & Opportunistic Science

The remaining domains are related. In a orienteering problem, the agent is given a starting location, and a set of checkpoints, each with some score associated with it. The agent seeks to visit these checkpoints and return to the starting location such that it maximizes the sum of the scores it accumulates while returning by some deadline (Vansteenwegen, Souffriau, and Oudheusden 2011). A situated agent in this setting must balance its short term goals to accumulate the most points with its long term constraint that it must return by the time limit. Opportunistic science can be seen as a relaxation of the orienteering problem. Rather than receiving knowledge of all the rewards at time zero, the agent gets transient opportunities to spend surplus resources in exchange for rewards. An example is a Mars rover that could observe a dust devil and record it, or use it to clean off its solar panels (Lorenz and Reiss 2015).

Conclusions

Situated planning, in contrast with traditional offline planning, has time progress while the agent plans. Prior work in situated planning has been mainly theoretical or domain independent. I aim to explore situated planning in three specific contrasting domains, giving us wide view of how the theory of situated planning can be applied. Our work so far has been focused on the grid path planning domain, which has shown some promising initial results.

References

Cashmore, M.; Coles, A.; Cserna, B.; Karpas, E.; Magazzeni, D.; and Ruml, W. 2018. Temporal planning while the clock ticks. *ICAPS*.

Dionne, A. J.; Thayer, J. T.; and Ruml, W. 2011. Deadline-Aware Search Using On-line Measures of Behavior. *SOCS*.

Foschini, L.; Hershberger, J.; and Suri, S. 2014. On the Complexity of Time-Dependent Shortest Paths. *Algorithmica*.

Lorenz, R. D.; and Reiss, D. 2015. Solar panel clearing events, dust devil tracks, and in-situ vortex detections on Mars. *Icarus*.

Phillips, M.; and Likhachev, M. 2011. SIPP: Safe interval path planning for dynamic environments. *ICRA*.

Russell, S.; and Wefald, E. 1991. Principles of metareasoning. *Artificial intelligence*.

Shperberg, S. S.; Coles, A.; Cserna, B.; Karpas, E.; Ruml, W.; and Shimony, S. E. 2019. Allocating planning effort when actions expire. *AAAI*.

Vansteenwegen, P.; Souffriau, W.; and Oudheusden, D. V. 2011. The orienteering problem: A survey. *EJOR*.