Situated Planning with Soft Goals
(Extended Abstract)

Andrew Coles[1], Erez Karpas[2], Shahaf S. Shperberg[3], Eyal Shlomo Shimony[3], Wheeler Ruml[4]
[1]King's College London, UK
[2]Technion, Israel
[3]Ben-Gurion University of the Negev, Israel
[4]University of New Hampshire, USA
andrew.coles@kcl.ac.uk, karpase@technion.ac.il shperbsh@post.bgu.ac.il, shimony@cs.bgu.ac.il, ruml@cs.unh.edu

Introduction

Unlike standard planning and search, situated planning (Cashmore et al. 2018) makes the more realistic assumption that if the agent spends x time on planning, then it can start executing its plan no earlier than at time x. The time required for planning is not known in advance and different plans may have different *latest possible start times*, thus necessitating metareasoning (Shperberg et al. 2019, 2021).

By providing the first situated planner that can handle soft goals, this paper significantly extends the reach of situated planning toward realistic problems. In prior work on situated planning (Shperberg et al. 2021), the planner stops after finding the first plan, as it needs only to find a timely one. However, in our setting, the planner may be better off continuing to search for a better plan.

The OPTIC planner (Benton, Coles, and Coles 2012), which is a baseline for this paper, supports planning with soft goals and non-instantaneous 'durative' actions (Fox and Long 2003), but still follows the standard planning setting, assuming that execution starts at time 0.

Our work uses the heuristics developed in prior work on planning with soft goals and on the metareasoning search strategies for situated planning. Most relevant to our work is Coles and Coles (2013), which derives multiple heuristic estimates from a relaxed plan rp built to all reachable soft goals, reflecting the trade-off between distance-to-go and the cost that will be reached. A relaxed plan $rp' \subset rp$ will have a lower distance-to-go, but a higher cost, as it reaches fewer soft goals (i.e., violates more soft goals).

Situated Planning with Soft Goals

As in related work (Shperberg et al. 2019, 2021; Coles et al. 2024), we view situated planning as a sequence of resource allocation decisions under uncertainty. In a situated setting, the time spent planning is a cost that must be weighed against the potential improvement in plan quality.

At each step, the planner must decide which partial plan (node) i to expand next to maximize the expected utility of the final outcome. To address this, we propose *Soft Goal-Aware Metareasoning Estimates (SGAME)*. Previous situated planning approaches typically maintained a single estimate for every partial plan. The core innovation of SGAME is maintaining a *list of estimates* for each partial plan, where each list item captures the trade-off of a specific subset of soft goals. Specifically, relying on the underlying planner's ability to distinguish between different nondominated goal subsets, we characterize each partial plan i by maintaining:

- A *list* M_i of Cumulative Distribution Functions (CDFs), where $M_i[l]$ estimates the probability distribution of the *additional* search time required to refine the partial plan into a valid plan achieving the l-th subset of soft goals.
- A *list* c_i of projected costs, where $c_i[l]$ is the estimated cost for that outcome. Our formulation defines cost as the utility of unachieved soft goals (i.e., utility loss).
- A *list* d_i of deadlines, where $d_i[l]$ represents the latest time by which execution must start to remain valid.

The metareasoning module uses these estimates to estimate an *optimal time allocation*. To do this, we maintain a global Pareto frontier Π of already discovered plans, represented as a list of cost-deadline pairs (c_j, d_j) sorted by increasing deadline. We augment the frontier by a dummy plan with cost c_f and deadline ∞. c_f is the cost incurred if the planner fails to find any valid timely plan.

Example 1. Consider a simplified scenario where $c_f = 100$, and the planner can explore one of two partial plans (nodes) i_{safe} and i_{risky}, both with a deadline $d = 19$:

- Node i_{safe}: $M_{\text{safe}} = [10 : 1]$, guaranteed to find a plan after 10 time units with cost $c = 15$.
- Node i_{risky}: $M_{\text{risky}} = [10 : 0.5, \infty : 0.5]$, will find a plan of cost 5 after 10 time units, or fail with probability 0.5 (represented by returning a plan at time ∞).

Here time is too short to explore both nodes before their deadlines pass. **Scenario 1) No existing plans:** The expected cost of exploring i_{safe} is 15. The expected cost of exploring i_{risky} is $0.5(5) + 0.5(100) = 52.5$. The rational choice is to exlore only i_{safe}. **Scenario 2) Fallback plan with cost 20:** Suppose the planner already found planπ_{prev} with cost 20, to start no later than at time 20. Due to π_{prev}, the expected cost of exploring i_{safe} becomes $\min(15, 20) = 15$. The expected cost of exploring i_{risky} becomes $0.5(\min(5, 20)) + 0.5(20) = 2.5 + 10 = 12.5$. The rational choice switches to i_{risky}. This demonstrates how the presence of a fallback plan encourages risk-taking to achieve higher quality, a key feature of our situated approach.

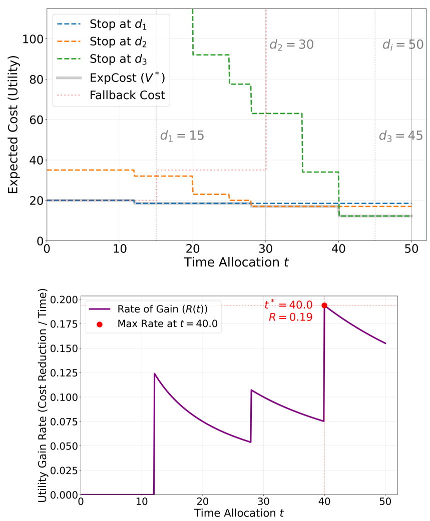

Figure 1: Visualization of `ExpCost` and `GreedyRate`

Since optimal metareasoning is intractable, SGAME is a greedy approximation guiding the search towards nodes that maximize the expected utility gain rate, inspired by the single-goal version (Shperberg et al. 2021). In SGAME, an `ExpCost` procedure answers the question: *"If we invest t more time units in this plan, what is the expected outcome?"*

Figure 1(top): expected cost (y-axis) vs. time allocated to search (x-axis) illustrates `ExpCost`. Dashed lines represent expected cost for a commitment to a deadline d_j from the Pareto frontier. We consider two existing plans: π_1 (blue) with cost $c_1 = 20$ and deadline $d_1 = 15$; and π_2 (orange) with $c_2 = 35$ and deadline $d_2 = 30$. Green represents failure: $c_f = 150$.Current partial plan i has projected cost $c_i = 5$, deadline $d_i = 50$. Its completion time is a distribution $M_i = [12 : 0.1, 20 : 0.4, 25 : 0.5, 28 : 0.6, 35 : 0.8, 40 : 0.95, 100 : 1.0]$. The last time point exceeds the deadline, so plan i is not guaranteed to be available on time. `ExpCost` identifies 3 distinct regions of optimality. For small t, the safest strategy is to stop at $d_1 = 15$, minimizing the risk of failure. Higher time allocations make finding a better plan more likely, thus best to aim for $d_2 = 30$. For even greater time allocations, increased success probability justifies an indefinite deadline (green curve), betting everything on completing plan i on time. The transparent black line tracks the minimum of these curves, demonstrating how `ExpCost` dynamically switches between strategies.

Second, `GreedyRate` computes a 'bang-for-buck' metric: the rate of utility gain per unit of computation time. It searches for a time allocation t that maximizes the improvement in expected value relative to the current best plan, normalized by the time invested. Figure 1(bottom) visualizes `GreedyRate` for the same scenario. The observed sawtooth pattern is due to discrete probability masses (jumps), with gradual decreases in regions where probability of termination remains constant, `GreedyRate` selects the global maximum $t^* = 40$ to maximize the efficiency of the search in terms of utility gain. The planner's priority function orders nodes by `GreedyRate`.

Alg.	Metric Baseline	Metric SGAME	Expansions Baseline	Expansions SGAME
inst#	ROVERS			
pfile5	3.8 (0.4)	**2.0 (0.0)**	357.0 (10.7)	92.6 (2.1)
pfile6	5.6 (0.5)	**3.4 (0.5)**	887.2 (38.1)	133.6 (5.2)
pfile7	3.4 (0.5)	**2.0 (0.0)**	362.6 (5.2)	2263.8 (66.1)
inst#	SATELLITE			
pfile3	**2.0 (0.0)**	3.0 (0.0)	283.8 (6.1)	647.8 (17.0)
pfile5	5.0 (0.0)	**3.0 (0.0)**	389.2 (3.2)	126.6 (31.9)
pfile18	11.6 (0.9)	**0.0 (0.0)**	887.2 (496.0)	108.0 (0.0)

Table 1: Results for Real Clock

Typical Experimental Results

We implemented SGAME in the OPTIC-based planner of Shperberg et al. (2021). We generated temporal problems with soft goals and deadlines by modifying existing problems in ROVERS, SATELLITE, and DRIVERLOG, from IPC-3. We assumed that one unit of time in the PDDL represents one second of real time. We compare SGAME to the baseline: the situated planner of (Shperberg et al. 2021) modified to keep searching for a better plan after finding the first plan.

Table 1(left) reports number of unachieved soft goals, denoted 'Metric', for some instances where they differ. Due to noise introduced by real-time runs, we report the average and standard deviation over 5 runs. As these results show, SGAME frequently does better than the baseline. Table 1(right) shows the number of expanded nodes for each problem. SGAME also tends to expand fewer nodes.

Acknowledgements

We are grateful for support from the US-Israel BSF (grant 2019730) and from the US NSF (grant 2008594).

References

Benton, J.; Coles, A.; and Coles, A. 2012. Temporal Planning with Preferences and Time-Dependent Continuous Costs. In *Proceedings of ICAPS*.

Cashmore, M.; Coles, A.; Cserna, B.; Karpas, E.; Magazzeni, D.; and Ruml, W. 2018. Temporal Planning while the Clock Ticks. In *Proceedings of ICAPS*.

Coles, A.; and Coles, A. 2013. Searching for Good Solutions in Goal-Dense Search Spaces. In *Proceedings of ICAPS*.

Coles, A.; Karpas, E.; Lavrinenko, A.; Ruml, W.; Shimony, S. E.; and Shperberg, S. S. 2024. Planning and Acting While the Clock Ticks. In *Proceedings of ICAPS*.

Fox, M.; and Long, D. 2003. PDDL2.1: An Extension to PDDL for Expressing Temporal Planning Domains. *JAIR*, 20.

Shperberg, S. S.; Coles, A.; Cserna, B.; Karpas, E.; Ruml, W.; and Shimony, S. E. 2019. Allocating Planning Effort When Actions Expire. In *Proceedings of AAAI*, 2371–2378.

Shperberg, S. S.; Coles, A.; Karpas, E.; Ruml, W.; and Shimony, S. E. 2021. Situated Temporal Planning Using Deadline-aware Metareasoning. In *Proc. ICAPS*, 340–348.