The PANDA λ System for HTN Planning in the 2023 IPC

Daniel Höller

Saarland University, Saarland Informatics Campus,
Saarbrücken, Germany
hoeller@cs.uni-saarland.de

Abstract

The PANDA λ system is an HTN planning system that can handle both totally ordered and partially ordered HTN models. It performs a progression search, i.e., it only processes tasks without predecessor in the task network and is based on the PANDA framework. PANDA λ uses a graph search and guides search by using a combination of heuristics and landmarks. These are combined by using a multi-fringe system.

Introduction

PANDA λ (Landmark-based PANDA) is a planning system from the PANDA framework (Höller et al. 2021), which can handle both totally ordered and partially ordered models.

Search-based systems in HTN planning can be divided into plan space-based systems and progression-based systems (see Bercher, Alford, and Höller, 2019). The latter only process tasks without predecessor in the task ordering of the current task network. PANDA λ is based on the systematic progression search introduced by Höller et al. (2020).

It uses the preprocessing stack of the PANDA framework: HDDL (Höller et al. 2020) as input language and by the grounding procedure introduced by Behnke et al. (2020).

During search, PANDA λ maintains a black-list of already visited nodes and processes every node only a single time, i.e., it uses a graph search. While this is (from a computational perspective) no problem in totally ordered HTN planning, it gets a task as hard as graph isomorphism in partially ordered HTN planning. To do it efficiently, PANDA λ uses the techniques introduced by Höller and Behnke (2021), which apply several techniques for hashing search nodes, and exploit certain special cases present in many models of the commonly used benchmark sets.

PANDA λ guides its search by using a combination of landmarks (Höller and Bercher 2021) and heuristics from the family of Relaxed Composition (RC) heuristics (Höller et al. 2018, 2019, 2020) to estimate the goal distance.

Similar to the LAMA system from classical planning (Richter and Westphal 2010), PANDA λ combines these in a multi-fringe search, where one fringe is sorted by a RC heuristic, and one by the LM-count heuristic computed on the landmarks. The system extracts nodes from the fringes in turn and each successor node is inserted into both fringes with the respective heuristic estimate.

We next describe the RC heuristics and the used landmarks afterwards. Each configuration of our overall system combines one of the two RC heuristics with one of the two landmark sets.

RC Heuristics

The family of RC heuristics (Höller et al. 2018, 2019, 2020) uses classical heuristics to estimate the goal distance during HTN search. To do so, it relaxes the HTN model to a classical model which is only used for heuristic calculation. It is created in a way that the set of solutions increases compared to the HTN model. HTN planning starts with the initial task(s) and decomposes them until only actions are left. This process can be seen as the building process of a tree. The classical RC model maintains which tasks are part of that tree, but in a bottom-up manner, *compositing* tasks. When an action from the original HTN is applied in this model, it is marked as part of the tree. Methods are represented in the RC model by special actions. These are applicable when all subtasks of the method are part of the tree. When they are applied, the decomposed task is marked as part of the tree. The goal of the overall problem is to mark the tasks in the current task network as being part of the tree.

This encoding solves several problems when translating HTN models to classical models. First, we always have a state-based goal (which is not the case in HTN models): adding the current tasks to the tree. Second, the model is also informed about applicability of actions, since actions can only be added when they are applicable. Like in other HTN heuristics, the encoding allows for task insertion (adding further actions apart from the decomposition hierarchy) to make actions applicable that are needed elsewhere. However, what is interesting about our encoding when compared to other heuristics (see e.g. Bercher et al., 2017), is that the costs of these added actions are incorporated into the heuristic value. In our implementation, we further restrict task insertion to those actions still reachable via decomposing the current task network. Third, our heuristic is – to some extend – informed about the decomposition process, because the tree must be created up to the current tasks.

Practically, the model can be updated instead of recomputed. The only things that need to be changed are the initial state and the goal condition of the RC model. The model is linear in the size of the HTN model, and can be combined with any classical heuristic. However, the update of the goal is not possible (efficiently) in every classical heuristic.

In the IPC, we combine it with the Add (Bonet and Geffner 2001) heuristic and with the FF (Hoffmann and Nebel 2001) heuristic.

Landmark Generation

In the IPC, we use two types of landmarks, RC-based and AND/OR landmarks, which are described in this section.

RC-based Landmarks

The first type of landmarks computes the LM-Cut heuristic (Helmert and Domshlak 2009) on the RC model of the initial search node. The generated landmarks are stored and tracked during search.

AND/OR Landmarks

The second type of landmarks is generated using the approach of Höller and Bercher (2021). It extends an approach from classical planning by Keyder, Richter, and Helmert (2010), who represent a delete-free classical planning problem as AND/OR graph, and extract landmarks from this graph afterwards. We extend the AND/OR graph to also represents parts of the decomposition hierarchy, and applies the unchanged extraction algorithm afterwards.

In contrast to the classical case, the HTN representation comes with more relaxations than only delete-relaxation. E.g., no ordering relations from the HTN model are represented in the graph.

We generate the landmarks on the initial search nodes and track them afterwards during search.

References

Behnke, G.; Höller, D.; Schmid, A.; Bercher, P.; and Biundo, S. 2020. On Succinct Groundings of HTN Planning Problems. In *Proceedings of the 34th AAAI Conference on Artificial Intelligence (AAAI)*, 9775–9784. AAAI Press.

Bercher, P.; Alford, R.; and Höller, D. 2019. A Survey on Hierarchical Planning – One Abstract Idea, Many Concrete Realizations. In *Proceedings of the 29th International Joint Conference on Artificial Intelligence (IJCAI)*, 6267–6275. IJCAI organization.

Bercher, P.; Behnke, G.; Höller, D.; and Biundo, S. 2017. An Admissible HTN Planning Heuristic. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence (IJCAI)*, 480–488. IJCAI organization.

Bonet, B.; and Geffner, H. 2001. Planning as heuristic search. *Artificial Intelligence*, 129(1-2): 5–33.

Helmert, M.; and Domshlak, C. 2009. Landmarks, Critical Paths and Abstractions: What's the Difference Anyway? In *Proceedings of the 19th International Conference on Automated Planning and Scheduling (ICAPS)*.

Hoffmann, J.; and Nebel, B. 2001. The FF Planning System: Fast Plan Generation Through Heuristic Search. *Journal of Artificial Intelligence Research*, 14: 253–302.

Höller, D.; and Behnke, G. 2021. Loop Detection in the PANDA Planning System. In *Proceedings of the 31st International Conference on Automated Planning and Scheduling (ICAPS)*. AAAI Press.

Höller, D.; Behnke, G.; Bercher, P.; and Biundo, S. 2021. The PANDA Framework for Hierarchical Planning. *Künstliche Intelligenz*, 30(1): 11–20.

Höller, D.; Behnke, G.; Bercher, P.; Biundo, S.; Fiorino, H.; Pellier, D.; and Alford, R. 2020. HDDL: An Extension to PDDL for Expressing Hierarchical Planning Problems. In *Proceedings of the 34th AAAI Conference on Artificial Intelligence (AAAI)*, 9883–9891. AAAI Press.

Höller, D.; and Bercher, P. 2021. Landmark Generation in HTN Planning. In *Proceedings of the 25th AAAI Conference on Artificial Intelligence (AAAI)*, 11826–11834. AAAI Press.

Höller, D.; Bercher, P.; Behnke, G.; and Biundo, S. 2018. A Generic Method to Guide HTN Progression Search with Classical Heuristics. In *Proceedings of the 28th International Conference on Automated Planning and Scheduling (ICAPS)*, 114–122. AAAI Press.

Höller, D.; Bercher, P.; Behnke, G.; and Biundo, S. 2019. On Guiding Search in HTN Planning with Classical Planning Heuristics. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence (IJCAI)*, 6171–6175. IJCAI organization.

Höller, D.; Bercher, P.; Behnke, G.; and Biundo, S. 2020. HTN Planning as Heuristic Progression Search. *Journal of Artificial Intelligence Research (JAIR)*, 67: 835–880.

Keyder, E.; Richter, S.; and Helmert, M. 2010. Sound and Complete Landmarks for And/Or Graphs. In *Proceedings of the 19th European Conference on Artificial Intelligence (ECAI)*, 335–340. IOS Press.

Richter, S.; and Westphal, M. 2010. The LAMA Planner: Guiding Cost-Based Anytime Planning with Landmarks. *Journal of Artificial Intelligence Research (JAIR)*, 39: 127–177.