TFTM-ArgMax Planner: Pruning Preferred Operators with Novelty

Michael Katz[1], **Alexander Tuisov**[2],

[1] IBM T.J. Watson Research Center, Yorktown Heights, USA
[2] Technion - Israel Institute of Technology, Haifa, Israel
michael.katz1@ibm.com, queldelan@gmail.com

Abstract

The planner *TFTM-ArgMax*, which stands for *The Fewer The Merrier* exploits red-black planning heuristic with a direct handling of conditional effects, using it as a base for a novelty heuristic, as well as using the novelty heuristic for pruning preferred operators. The preferred operators are pruned by choosing a subset of the preferred operators of underlying heuristic with the highest novelty score.

Introduction

Red-black planning (Katz, Hoffmann, and Domshlak 2013b,a; Katz and Hoffmann 2013, 2014; Domshlak, Hoffmann, and Katz 2015) allows to partially relax a planning task while remaining in a tractable fragment of planning. Recently, Katz (2019) has shown that the fragment of red-black planning characterized by *DAG black causal graphs* remains tractable in the presence of conditional effects, extending the existing red-black planning heuristics to natively handling conditional effects. This native support of conditional effects was integrated into the *Cerberus* planner (Katz 2018), which participated in IPC 2018. Another feature of *Cerberus* is the use of a search pruning technique based on the concept of *novelty* of a state, where the search procedure prunes nodes that do not qualify as *novel*. *Cerberus* exploits the novelty of a state with respect to its heuristic estimate (Katz et al. 2017). The notion was no longer used solely for pruning search nodes, but rather as a heuristic function, for node ordering in a queue. Since such heuristics are not goal-aware, *Cerberus* uses the base red-black heuristic as a secondary (tie-breaking) heuristic for node ordering. Following the success of the LAMA planner (Richter, Westphal, and Helmert 2011), the planner used an additional queue for successors achieved by preferred operators. While the use of prefer operators greatly improves planner performance, sometimes the large number of preferred operators can negatively impact performance, and even a random selection of a subset of these operators can have a significant positive effect on the overall performance (Tuisov and Katz 2021).

The planner *TFTM* uses novelty-based pruning of preferred operators to reduce the set considered by the search. In all other aspects, it mimics the *Cerberus* planner.

In addition, *TFTM* uses the h^2 mutex detection (Alcázar and Torralba 2015) while translating from PDDL to SAS$^+$, for the satisficing track variants. For agile tracks, our analysis indicates that switching off the h^2 mutex detection significantly improved performance. In what follows, we describe the configurations submitted to each of the tracks.

Satisficing Track

The planner runs iterative search with multiple queues, starting with GBFS and continuing to lazy weighted A^*, with diminishing weights, 5, 3, 2, 1, and continuing with the weight 1. The heuristics used are the novelty of the heuristic estimate (red-black heuristic where the red-black planning fragment was created by iteratively painting invertible variables red until the black causal graph becomes acyclic), as well as the landmark count heuristic, mirroring the configuration of *Cerberus* from IPC 2018. The difference is that the preferred operators of the novelty heuristic are computed by pruning the set of preferred operators of the underlying heuristic. The formal definitions for the novelty heuristics and the pruning method used are given below.

Agile Track

The configuration submitted to this track runs the first iteration of the configuration submitted to the satisficing track, differing from that configuration in one aspect only: the h^2 mutex detection was not applied for this track.

Novelty Pruning Of Preferred Operators

In what follows, we present the definitions of Tuisov and Katz (2021) and Katz et al. (2017) on the novelty-based pruning of preferred operators used in the configuration.

We start with the definition of the *novelty score of a fact*.

Definition 1 (heuristic novelty) *Given a heuristic function* $h : \mathcal{S} \mapsto \mathbb{R}^{0+}$ *and a search history* $\mathcal{H}$*, the* **novelty score of a fact** f *is defined as*

$$N(f, \mathcal{H}, h) = \begin{cases} \min\limits_{s \in \mathcal{H}(f)} h(s), & \mathcal{H}(f) \neq \emptyset \\ \infty, & \textit{otherwise}. \end{cases}$$

Given a state s*, the* **novelty score of a fact** f **in state** s *is defined as* $N(f, s, \mathcal{H}, h) = N(f, \mathcal{H}, h) - h(s)$ *if* $f \in s$.

A search history $\mathcal{H}$ is a set of pairs of operators and states that these operators lead to, and $\mathcal{H}(f)$ is the set of states in the search history that contain the fact f. To simplify the notation, we sometimes do not mention the search history $\mathcal{H}$ and the heuristic h when these are clear from the context. A fact is novel in state s if its novelty score in s is strictly positive. A state is novel if it contains at least one novel fact.

Katz et al. (2017) define a variety of novelty based heuristics, starting with the most basic one, h_{BN}, separating novel states (that obtain the value 0) from the non-novel states (that obtain the value 1). The second heuristic function $h_{QN}(s) := |\mathcal{V}| - \sum_{f\in s} N^{+}(f,s)$ also separates novel states, based on the number of novel facts ($N^{+}(f,s)$ is 1 when $N(f,s) > 0$ and 0 otherwise). Finally, h_{QB} also separates non-novel states, based on the number of strictly non-novel facts.

$$h_{QB}(s) = \begin{cases} h_{QN}(s), & h_{QN}(s) < |\mathcal{V}| \\ |\mathcal{V}| + \sum_{f\in s} N^{-}(f,s), & \text{otherwise.} \end{cases}$$

While Katz et al. (2017) define additional heuristics, h_{QB} was found to be best performing overall in their experiments and is the novelty heuristic used by TFTM-ArgMax.

We define now *heuristic novelty* of operators, analogously to how a novelty of a fact is defined (Katz et al. 2017), see Definition 1.

Definition 2 (operator novelty score) *Given a heuristic function $h : \mathcal{S} \mapsto \mathbb{R}^{0+}$ and a search history $\mathcal{H}$, the* **novelty score of an operator** *o is defined as*

$$N(o,\mathcal{H},h) = \begin{cases} \min_{s\in\mathcal{H}(o)} h(s), & \mathcal{H}(o) \neq \emptyset \\ \infty, & \textit{otherwise}. \end{cases}$$

Further, given a state s, the **novelty score of an operator** *o* **in state** *s is defined as $N(o,s,\mathcal{H},h) = N(o,\mathcal{H},h) - h(s)$.*

In words, the novelty score of an operator in a state is the difference between the (best) heuristic value of a state previously reached by the operator during search and the heuristic value of the current state.

Finally, we formally define preferred operators for the novelty heuristic.

Definition 3 (max-novel preferred operators) *Given a heuristic function h, the max-novel preferred operators of h are defined as*

$$\mathcal{PO}_{\max}(s,\mathcal{H}) = \operatorname*{arg\,max}_{o\in\mathcal{PO}_h(s)} \{N(o,s,\mathcal{H},h)\}$$

The planner TFTM-ArgMax sets the operators $\mathcal{PO}_{\max}(s,\mathcal{H})$ as preferred, a configuration that was found to work well by Tuisov and Katz (2021).

Post-IPC Analysis

International Planning Competition (IPC) 2023 introduced 7 domains: *folding*, *labyrinth*, *quantum-layout*, *recharging-robots*, *ricochet-robots*, *rubiks-cube*, and *slitherlink*, with 20 instances in each. Here, we present some observations about planners behavior on these domains.

First, note that the translator component used by the planner is used by both agile and satisficing variants, while the preprocessing component (h^2 mutex detection) is used by the satisficing variant only. Translator fails on 16 instances of *labyrinth*, 3 instances of *recharging-robots*, and all 20 instances of *slitherlink*. On additional 12 instances of *recharging-robots* the translator creates axioms, which are not supported by the search component. In these cases, axioms are avoidable. The preprocessor fails on the remaining 4 instances of *labyrinth* and 5 instances of *folding*.

Second, the red-black heuristic (Domshlak, Hoffmann, and Katz 2015) seems to work as intended in most cases. Red-black heuristic extends the FF (Hoffmann and Nebel 2001) heuristic by considering delete effects of RSE-invertible variables (Domshlak, Hoffmann, and Katz 2015). Such variables are found in all domains where search could start, except for *rubiks-cube*. In the latter domain, the red-black heuristic values returned were essentially equivalent to the FF heuristic ones.

Third, the novelty-based pruning of preferred operators does not seem to pay off, comparing to not pruning the preferred operators, as is done by the *Cerberus* planner (Katz 2023). Specifically, in *rubiks-cube*, the pruning seems to be detrimental, reducing the coverage from 19 to 7.

Conclusions

The domains introduced in IPC 2023 are significantly different from the previously existing ones. In order to be able to efficiently handle tasks in these domains, the planner should be adapted to use a more efficient translator and preprocessor. An in-depth investigation of preferred operators behavior on these domains is in order.

References

Alcázar, V.; and Torralba, Á. 2015. A Reminder about the Importance of Computing and Exploiting Invariants in Planning. In Brafman, R.; Domshlak, C.; Haslum, P.; and Zilberstein, S., eds., *Proceedings of the Twenty-Fifth International Conference on Automated Planning and Scheduling (ICAPS 2015)*, 2–6. AAAI Press.

Domshlak, C.; Hoffmann, J.; and Katz, M. 2015. Red-Black Planning: A New Systematic Approach to Partial Delete Relaxation. *Artificial Intelligence*, 221: 73–114.

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

Katz, M. 2018. Cerberus: Red-Black Heuristic for Planning Tasks with Conditional Effects Meets Novelty Heuristic and Enchanced Mutex Detection. In *Ninth International Planning Competition (IPC-9): Planner Abstracts*, 47–51.

Katz, M. 2019. Red-Black Heuristic for Planning Tasks with Conditional Effects. In *Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence (AAAI 2019)*. AAAI Press.

Katz, M. 2023. Cerberus Planner: Back In Action. In *Tenth International Planning Competition (IPC-10): Planner Abstracts*.

Katz, M.; and Hoffmann, J. 2013. Red-Black Relaxed Plan Heuristics Reloaded. In Helmert, M.; and Röger, G., eds., *Proceedings of the Sixth Annual Symposium on Combinatorial Search (SoCS 2013)*, 105–113. AAAI Press.

Katz, M.; and Hoffmann, J. 2014. Pushing the Limits of Partial Delete Relaxation: Red-Black DAG Heuristics. In *ICAPS 2014 Workshop on Heuristics and Search for Domain-independent Planning (HSDIP)*, 40–44.

Katz, M.; Hoffmann, J.; and Domshlak, C. 2013a. Red-Black Relaxed Plan Heuristics. In desJardins, M.; and Littman, M. L., eds., *Proceedings of the Twenty-Seventh AAAI Conference on Artificial Intelligence (AAAI 2013)*, 489–495. AAAI Press.

Katz, M.; Hoffmann, J.; and Domshlak, C. 2013b. Who Said We Need to Relax *All* Variables? In Borrajo, D.; Kambhampati, S.; Oddi, A.; and Fratini, S., eds., *Proceedings of the Twenty-Third International Conference on Automated Planning and Scheduling (ICAPS 2013)*, 126–134. AAAI Press.

Katz, M.; Lipovetzky, N.; Moshkovich, D.; and Tuisov, A. 2017. Adapting Novelty to Classical Planning as Heuristic Search. In Barbulescu, L.; Frank, J.; Mausam; and Smith, S. F., eds., *Proceedings of the Twenty-Seventh International Conference on Automated Planning and Scheduling (ICAPS 2017)*, 172–180. AAAI Press.

Richter, S.; Westphal, M.; and Helmert, M. 2011. LAMA 2008 and 2011 (planner abstract). In *IPC 2011 Planner Abstracts*, 50–54.

Tuisov, A.; and Katz, M. 2021. The Fewer the Merrier: Pruning Preferred Operators with Novelty. In Zhou, Z.-H., ed., *Proceedings of the 30th International Joint Conference on Artificial Intelligence (IJCAI 2021)*, 4190–4196. IJCAI.