QDom-LMCut: Enhancing Search with Quantitative Dominance Pruning

Álvaro Torralba

Aalborg University, Denmark
alto@cs.aau.dk

Abstract

We present QDom-LMCut, an entry to the optimal track of the International Planning Competition 2023. QDom-LMCut explores the use of quantiative dominance analysis to reduce the search effort by pruning states during the search if they are worse than others.

Introduction

The core idea of QDom-LMCut is to reduce the search effort by performing quantitative dominance pruning (Torralba and Hoffmann 2015; Torralba 2017, 2021). The planner analyzes the task in order to establish a quantitative dominance function (QDF) that can bound the difference in goal distance between any two states in the task. The planner, then leverages the QDF to reduce search effort in two ways. On the one hand, it uses dominance pruning, removing from the search any node if a better alternative has already been seen during the search. On the other hand, action selection directly applies an action in a state if this is guaranteed to start an optimal plan. In domains where the quantitative analysis finds an informative QDF, this can reduce search effort by several orders of magnitude.

As search algorithm, we use A*with the LM-Cut heuristic (Helmert and Domshlak 2009). This is a very informative heuristic, although not fully competitive with state-of-the-art algorithms such as Scorpion (Seipp 2018) or Complementary (Franco et al. 2017; Franco, Lelis, and Barley 2018).

QDom: Quantitative Dominance Analysis

QDom-LMCut is implemented on top of the Fast Downward Planning System (Helmert 2006), and uses the h^2 preprocessor to simplify the task before planning (Alcázar and Torralba 2015).

Dominance Analysis

Before starting the search, QDom-LMCut analyzes the planning task in order to compute a quantitative dominance function (Torralba 2017).

To do so, first we use the merge-and-shrink algorithm (Helmert, Haslum, and Hoffmann 2007; Helmert et al. 2014; Sievers and Helmert 2021) to obtain a set of transition systems that represents the planning task. Specifically, we merge factors following the DFP merge strategy (Dräger, Finkbeiner, and Podelski 2006; Sievers, Wehrle, and Helmert 2014) until no more factors can be merged with 10 000 transitions. Note that, even though this is not considered the best merge strategy for deriving admissible heuristics (Sievers, Wehrle, and Helmert 2016), it can achieve good decompositions under which to compute the dominance function.

Furthermore, within the merge-and-shrink algorithm, we simplify the representation using label reduction (Sievers, Wehrle, and Helmert 2014), pruning subsumed transitions (Torralba and Kissmann 2015), using bisimulation shrinking (Helmert et al. 2014), and pruning unreachable and dead-end states.

Leveraging Quantitative Dominance Functions

We leverage QDFs in two ways during the search:

Dominance Pruning We compare each state t against all previously expanded states S. If the dominance function can prove that there exists some $s \in S$ that dominates t by being at least as close to the goal and reachable with lower or equal cost f, then we can discard t. To efficiently compare against all previously expanded states, we keep a representation of the set of states dominated by any expanded state, in the form of a Binary Decision Diagram (Bryant 1986). As this may be expensive, it only pays off if there is enough pruning. Therefore, after performing 100 expansions we disable pruning unless at least 30% of the nodes have been pruned.

Action Selection During successor generation, we compare each successor against its parent, by taking into account the effects of the action being applied. If the successor is dominated by the parent, then it is pruned, as it cannot possibly start an optimal plan from the parent. Otherwise, if the successor dominates the parent by an amount equal to the cost of the action being applied, then the successor is guaranteed to start an optimal plan. In that case, the successor is kept and all other successors are automatically pruned.

Conclusion

We have introduced QDom-LMCut, a planner that features dominance analysis in the IPC'23. While the heuristic function is not fully competitive with the state of the art nowadays, the objective of the planner is to showcase dominance pruning techniques as part of the competition.

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.

Bryant, R. E. 1986. Graph-Based Algorithms for Boolean Function Manipulation. *IEEE Transactions on Computers*, 35(8): 677–691.

Dräger, K.; Finkbeiner, B.; and Podelski, A. 2006. Directed Model Checking with Distance-Preserving Abstractions. In Valmari, A., ed., *Proceedings of the 13th International SPIN Workshop (SPIN 2006)*, volume 3925 of *Lecture Notes in Computer Science*, 19–34. Springer-Verlag.

Franco, S.; Lelis, L. H. S.; and Barley, M. 2018. The Complementary2 Planner in the IPC 2018. In *Ninth International Planning Competition (IPC-9): Planner Abstracts*, 32–36.

Franco, S.; Torralba, Á.; Lelis, L. H. S.; and Barley, M. 2017. On Creating Complementary Pattern Databases. In Sierra, C., ed., *Proceedings of the 26th International Joint Conference on Artificial Intelligence (IJCAI 2017)*, 4302–4309. IJCAI.

Helmert, M. 2006. The Fast Downward Planning System. *Journal of Artificial Intelligence Research*, 26: 191–246.

Helmert, M.; and Domshlak, C. 2009. Landmarks, Critical Paths and Abstractions: What's the Difference Anyway? In Gerevini, A.; Howe, A.; Cesta, A.; and Refanidis, I., eds., *Proceedings of the Nineteenth International Conference on Automated Planning and Scheduling (ICAPS 2009)*, 162–169. AAAI Press.

Helmert, M.; Haslum, P.; and Hoffmann, J. 2007. Flexible Abstraction Heuristics for Optimal Sequential Planning. In Boddy, M.; Fox, M.; and Thiébaux, S., eds., *Proceedings of the Seventeenth International Conference on Automated Planning and Scheduling (ICAPS 2007)*, 176–183. AAAI Press.

Helmert, M.; Haslum, P.; Hoffmann, J.; and Nissim, R. 2014. Merge-and-Shrink Abstraction: A Method for Generating Lower Bounds in Factored State Spaces. *Journal of the ACM*, 61(3): 16:1–63.

Seipp, J. 2018. Fast Downward Scorpion. In *Ninth International Planning Competition (IPC-9): Planner Abstracts*, 77–79.

Sievers, S.; and Helmert, M. 2021. Merge-and-Shrink: A Compositional Theory of Transformations of Factored Transition Systems. *Journal of Artificial Intelligence Research*, 71: 781–883.

Sievers, S.; Wehrle, M.; and Helmert, M. 2014. Generalized Label Reduction for Merge-and-Shrink Heuristics. In Brodley, C. E.; and Stone, P., eds., *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence (AAAI 2014)*, 2358–2366. AAAI Press.

Sievers, S.; Wehrle, M.; and Helmert, M. 2016. An Analysis of Merge Strategies for Merge-and-Shrink Heuristics. In Coles, A.; Coles, A.; Edelkamp, S.; Magazzeni, D.; and Sanner, S., eds., *Proceedings of the Twenty-Sixth International Conference on Automated Planning and Scheduling (ICAPS 2016)*, 294–298. AAAI Press.

Torralba, Á. 2017. From Qualitative to Quantitative Dominance Pruning for Optimal Planning. In Sierra, C., ed., *Proceedings of the 26th International Joint Conference on Artificial Intelligence (IJCAI 2017)*, 4426–4432. IJCAI.

Torralba, Á. 2021. On the Optimal Efficiency of A* with Dominance Pruning. In Leyton-Brown, K.; and Mausam, eds., *Proceedings of the Thirty-Fifth AAAI Conference on Artificial Intelligence (AAAI 2021)*, 12007–12014. AAAI Press.

Torralba, Á.; and Hoffmann, J. 2015. Simulation-Based Admissible Dominance Pruning. In Yang, Q.; and Wooldridge, M., eds., *Proceedings of the 24th International Joint Conference on Artificial Intelligence (IJCAI 2015)*, 1689–1695. AAAI Press.

Torralba, Á.; and Kissmann, P. 2015. Focusing on What Really Matters: Irrelevance Pruning in Merge-and-Shrink. In Lelis, L.; and Stern, R., eds., *Proceedings of the Eighth Annual Symposium on Combinatorial Search (SoCS 2015)*, 122–130. AAAI Press.