

BDDs Strike Back (in AI Planning)

Stefan Edelkamp

Institute of Artificial Intelligence
University of Bremen
edelkamp@tzi.de

Peter Kissmann and Álvaro Torralba

Foundations of Artificial Intelligence
Saarland University, Saarbrücken
{kissmann,torralba}@cs.uni-saarland.de

Abstract

The cost-optimal track of the international planning competition in 2014 has seen an unexpected outcome. Different to the precursing competition in 2011, where explicit-state heuristic search planning scored best, advances in the state-set exploration with BDDs showed a significant lead. In this paper we review the outcome of the competition, briefly looking into the internals of the competing systems.

Introduction

The deterministic part of the International Planning Competition (IPC) runs every 2-3 years. In 2014¹, it included a temporal, a sequential, and an agile track, optimizing different criteria for plans (action cost, CPU time and total-time). The PDDL input specification language has not been changed, but more emphasis was given to conditional effects.

Probably most exiting was the cost-optimal sequential planning track, enforcing plans of minimal sum of action costs (see Fig. 1). The top five performing planners were SYMBA*-2 (151 of 280 problems solved), SYMBA*-1 (143), cGAMER (120), SPM&S (114), RIDA* (113), DYNAMIC-GAMER (99). All but one of these systems exploit (reduced ordered) binary decision diagrams (BDDs) for state-space traversal and heuristic guidance (Edelkamp and Kissmann 2009).

While in 2008² the BDD-based planner GAMER won the sequential optimal track, in 2011 explicit-state heuristic search planners, especially portfolio planners took the lead³. Among the leading heuristics in the portfolios were *LM-cut* and *Merge&Shrink*. Many competitors in 2014, thus, were focused on the combination of multiple planners or heuristics. There were also new methods like flow heuristics, automated pruning via symmetries and partial-order reduction or incremental computation of *LM-cut* (Chrapa, Vallati, and McCluskey 2014). But this time, BDDs stroke back.

Copyright © 2015, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

¹<http://helios.hud.ac.uk/scommv/IPC-14>

²<http://ipc.informatik.uni-freiburg.de>

³<http://www.plg.inf.uc3m.es/ipc2011-deterministic>

BDD Basics

What is a BDD? A BDD is a directed acyclic graph data structure for a Boolean function (Wegener 2000). Nodes are labeled with variables, edges and sinks are labeled with 1 and 0. Redundant nodes are eliminated and the variable ordering is the same on every path.

What do BDDs represent? BDDs are characteristic functions of planning state sets (Edelkamp 2014). Each path from root to the 1-sink acts as the binary representation of one state in the set. Initial and goal conditions can also be represented as such state sets.

How does planning with BDDs work? Planning actions are cast as a set representation of their pre- and post-conditions (transition relation). In the *image* symbolic exploration checks the pre- and applies the postcondition (Edelkamp, Kissmann, and Torralba 2012a).

What is the advantage of BDDs? Firstly, BDDs can have an advantage in space. Some polynomial-sized BDDs represent exponentially many states (Edelkamp and Kissmann 2008). Secondly, forward and backward exploration are the same, except that the initial and goal condition as well as the pre- and postcondition are exchanged. Thirdly, the compact representation of many states in a smaller structure, often results in faster runtimes.

Can BDDs execute heuristic search? Algorithms like BDDA* (Edelkamp and Reffel 1999) use BDDs for the search space and for the heuristic. Symbolic PDB heuristics (Edelkamp 2002) refer to backward unguided exploration in abstract state spaces. Other heuristics like *Merge&Shrink* also enjoy a BDD representation (Edelkamp, Kissmann, and Torralba 2012b).

Essentials of the Top IPC Performers

In the BDD-based planners of the 2014 competition, problems in (pre-) processing apparent in some of the 2011 domains have been resolved and the support of conditional effects has been added. New partitioned ways of computing the successor set were applied (Torralba, Edelkamp, and Kissmann 2013). Mutex (h^2) constraints were used to simplify the problem and rule out illegal states (Torralba and Alcázar 2013). On top of these advances, the best performers have the following ingredients.

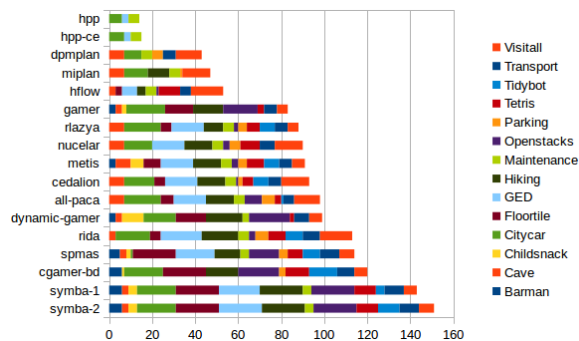


Figure 1: IPC-2014 Results, Sequential Optimal Track.

- GAMER variants used symbolic bidirectional blind search: DYNAMIC-GAMER uses dynamic variable re-ordering during the search (Kissmann and Hoffmann 2013) and CGAMER extends GAMER with state-invariant constraints and improved successor generation methods.
- SYMBA* is a combination of forward and backward symbolic heuristic searches. It uses perimeter abstraction heuristics (Dillenburg and Nelson 1994). for the search frontiers to meet and to finally prove optimality.
- SPM&S applies A* search with symbolic perimeter abstraction heuristics. It performs symbolic regression to construct a perimeter around the goal, which is used to initialize a symbolic version of PDBs and *Merge&Shrink* heuristics (Torralba, Linares López, and Borrajo 2013).
- RIDA* is a non-symbolic planner that combines different heuristics, selecting a subset of useful heuristics for each given problem (Barley, Franco, and Riddle 2014). Among other heuristics, such as *LM-cut*, it uses genetic algorithms to generate a large set of PDBs. A gene representation of variable selection patterns is populated and evolution selects the fittest. Note that the original work on the automated selection of PDB heuristics with genetic algorithms already used BDDs (Edelkamp 2007).

Lessons Learnt

The outcome of a planning competition is clearly subject to the benchmark domains chosen. We do not expect a universal best solver across all domains, but the comeback of BDDs in AI planning in 2014 is remarkable. With the rise of more efficient SAT solvers and the success story of heuristic search planners, BDDs were no longer seen as first choice in state-space planning. The results of IPC 2014, however, show that this interpretation could be revised.

BDDs are especially good in the construction of PDBs and related estimates. They support breadth-first and (discrete) cost-first search in both directions (Kissmann and Edelkamp 2011). The top IPC 2014 planners highlight the importance of regression and bidirectional search. SYMBA*, CGAMER and DYNAMIC-GAMER additionally applies symbolic bidirectional search and SPM&S uses standard A* search, but symbolic regression to generate a perimeter around the goal as well as abstraction heuristics.

Apart from symbolic search planners, in IPC 2014 most planners have been focused on different methods for combining heuristics, either portfolio design approaches or utility-estimation based methods. An important factor for the success of such methods is the diversity of the estimate values. As the experiments in RIDA* show, automated methods for the combination of PDB heuristic estimates that exploit this characteristic are superior to the state-of-the-art.

Acknowledgments Thanks to all our team members, the organizers, the competitors, and the supporters of IPC 2014.

References

- Barley, M. W.; Franco, S.; and Riddle, P. J. 2014. Overcoming the utility problem in heuristic generation: Why time matters. In *ICAPS*.
- Chrapa, L.; Vallati, M.; and McCluskey, L. 2014. The 2014 International Planning Competition. Description of participating planners.
- Dillenburg, J. F., and Nelson, P. C. 1994. Perimeter search. *Artif. Intell.* 65(1):165–178.
- Edelkamp, S., and Kissmann, P. 2008. Limits and possibilities of BDDs in state space search. In *AAAI*.
- Edelkamp, S., and Kissmann, P. 2009. Optimal symbolic planning with action costs and preferences. In *IJCAI*.
- Edelkamp, S., and Reffel, F. 1999. OBDDs in heuristic search. In *KI*, 81–92.
- Edelkamp, S.; Kissmann, P.; and Torralba, Á. 2012a. Lex-partitioning: A new option for BDD search. In *GRAPHITE*.
- Edelkamp, S.; Kissmann, P.; and Torralba, Á. 2012b. Symbolic A* search with pattern databases and the merge-and-shrink abstraction. In *ECAI*, 306–311.
- Edelkamp, S. 2002. Symbolic pattern databases in heuristic search planning. In *AIPS*, 274–293.
- Edelkamp, S. 2007. Automated creation of pattern database search heuristics. In *MOCHART*, 36–51.
- Edelkamp, S. 2014. Planning pattern databases. In *ECP 2001, Reprint*. AAAI Press.
- Kissmann, P., and Edelkamp, S. 2011. Improving cost-optimal domain-independent symbolic planning. In *AAAI*.
- Kissmann, P., and Hoffmann, J. 2013. What’s in it for my BDD? On causal graphs and variable orders in planning. In *ICAPS*.
- Torralba, Á., and Alcázar, V. 2013. Constrained symbolic search: On mutexes, BDD minimization and more. In *SOCS*.
- Torralba, Á.; Edelkamp, S.; and Kissmann, P. 2013. Transition trees for cost-optimal symbolic planning. In *ICAPS*.
- Torralba, Á.; Linares López, C.; and Borrajo, D. 2013. Symbolic merge-and-shrink for cost-optimal planning. In *IJCAI*.
- Wegener, I. 2000. *Branching Programs and Binary Decision Diagrams: Theory and Applications*. SIAM.