Symmetry Breaking: Satisficing Planning and Landmark Heuristics
ICAPS-2013 Doctoral Consortium

Alexander Shleyfman
Advisor: Prof. Carmel Domshlak
Faculty of Industrial Engineering and Managment
Technion - Israel Institute of Technology

Abstract

In heuristic search planning, state-space symmetries are mostly ignored by both the search algorithm and the heuristic guidance. Only recently, Pochter, Zohar, and Rosenschein (2011) introduced an effective framework for detecting and accounting for state symmetries within A^* cost-optimal planning. We extend this framework in order to allow usage of strictly larger symmetry groups. This enables more efficient pruning of strictly larger parts of the search space. Our approach is based on exploiting information about the parts of the transition system being gradually revealed by the search algorithms, such as A^*, WA^* or even *GBFS*.

In our work we will expand the use of cost-optimal planning by Pochter, Zohar, and Rosenschein (2011) to forward state-space search planning, and present two implementations of this framework using different pruning algorithms. Our aim is to evaluate their effectiveness both theoretically and practically.

Searching for computational tools that can further push the boundary of forward search planning, we show that addtional state-space symmetries techniques can substantially improve even the most effective forward heuristic-search planners, with respect to all standard performance measures. The improvement comes from the state-space and operator pruning, as well as from transparent cost-to-state updates and heuristics enhancement by information obtained during the search at different symmetric states.

An extensive empirical evaluation shows that our approach permits substantial reductions in search effort, which allows us to solve larger number of problems.

Introduction

Over the last two decades, the combined machinery of relaxation heuristics, preferred operators, and various enhancements of the very search infrastructure, have positioned heuristic forward search as a leading technique for classical planning, in terms of both efficiency and robustness.

Numerous admissible and non-admissible heuristics for domain-independent planning have been proposed, varying from cheap to compute and not very informative to expensive to compute and very informative (Bonet and Geffner 1998; 2001; Haslum and Geffner 2000; Hoffmann and Nebel 2001; Helmert 2004; Helmert, Haslum, and Hoffmann 2007; Katz and Domshlak 2010; Karpas and Domshlak 2009; Helmert and Domshlak 2009; Bonet and Helmert 2010; Richter and Westphal 2010). However, while further progress in developing informative heuristics is still very much desired, it is also well known that, on many problems both cost-optimal and satisficing searches expand an exponential number of nodes even if equipped with heuristics that are almost perfect in their estimates (Helmert and Röger 2008). One major reason for that is state symmetries in the transition systems of interest. A succinct description of the planning tasks in languages such as STRIPS and SAS^+almost unavoidably results in lots of different states in the search space to be symmetric to one another with respect to the task at hand. In turn, failing to detect and account for these symmetries results in forward searches as A^*, WA^* or *GBFS* searching through many symmetric states, although searching through a state is equivalent to searches through all of its symmetric counterparts.

The idea of identifying and pruning symmetries while reasoning about automorphisms of the search spaces has been exploited for quite a while already in model checking (Emerson and Sistla 2011), constraint satisfaction (Puget 1993), and planning (Rintanen 2003; Fox and Long 1999; 2002). However, until the recent work by Pochter et al. (2011), no empirical successes in this direction have been reported in the scope of cost-optimal planning as heuristic forward search. The success of the framework proposed by Pochter et al. is especially valuable because, to date, heuristic forward search with A^* constitutes the most effective approach to cost-optimal planning.

In this work, we build upon the framework of Pochter et al. (2011) and extend and improve it to allow for exploiting strictly larger sets of automorphisms, and thus pruning strictly larger parts of the search space. Our approach is based on exploiting information about the part of the transition system that is gradually being revealed by forward search algorithms as A^*. This information allows us to eliminate the requirement of Pochter et al. from the automorphisms to stabilize the initial state, a requirement that turns out to be quite constraining in terms of state-space pruning. We introduce a respective extension of the A^* algorithm that preserves its core properties of completeness and optimality. Similarly to the work of Pochter et al. , our approach works at the level of the search algorithm, and is completely independent of the heuristic in use.

Our empirical evaluation shows that our approach to A^*

symmetry breaking favorably competes with the previous work of Pochter et al. (2011), increasing the number of problems solved, and significantly reducing the search effort required to solve planning tasks.

Furthermore we will generalize this approach to forward search algorithms as *GBFS* and *WA**.

A prominent example of progress in the subfield of satisficing planning is LAMA-11 (or LAMA, for short), a heuristic-search planning system that won the sequential satisficing track of the International Planning Competition (IPC) in 2011 (Richter, Westphal, and Helmert 2011), with its predecessor, LAMA-08, winning the respective IPC track in 2008. LAMA builds on the Fast Downward system (Helmert 2006), inheriting the general structure of Fast Downward, the translation of propositional PDDL tasks to representations with finite-domain variables, and the exploitation of several heuristics simultaneously via a multi-queue search architecture. The two core features of LAMA are its iterated search using restarts (Richter, Thayer, and Ruml 2010), and the use of relaxation landmarks for defining heuristic estimates and preferred operators (Richter, Helmert, and Westphal 2008).

State-of-the-art planners these days carefully balance between search completeness and focus, between the informativeness of the heuristics and the cost of computing them. A very interesting question is what additional techniques can further stratify satisficing heuristic-search planning, either in terms of coverage or in terms of plan quality, or both? While this question is broad enough to have many positive answers, with the years it is getting harder and harder to push the boundary of satisficing planning. We will investigate prospects of reasoning about state-space symmetries within satisficing heuristic-search planning. Using the framework of goal-stable automorphisms for cost-optimal planning with A^* (Domshlak, Katz, and Shleyfman 2012), we show that its simple adaptation to greedier search procedures results in substantial state pruning, as well as in transparent improvement of discovered plan quality. Furthermore, we show that goal-stable automorphism groups such as those of Domshlak, Katz, and Shleyfman (2012) can be used to improve informativeness of landmark heuristic estimates, by aggregating information obtained during the search at different symmetric states. We show that both these features are very cost-effective, in the sense of robust improvement of both standard *GBFS* and LAMA's iterative search, with respect to all standard performance measures.

Background

We consider planning tasks $\Pi = \langle V, O, s_0, G, cost\rangle$ captured by the standard $\textsc{sas}^+$ formalism (Bäckström and Klein 1991; Bäckström and Nebel 1995) with operator costs. V is a set of finite-domain state variables, $S = \prod_{v \in V} dom(v)$ is the state space of Π, s_0 is an initial state, and goal G is a partial assignment to V; a state s is a goal state, denoted by $s \in S_*$, iff $G \subseteq s$. O is a finite set of operators, each given by a pair $\langle \mathsf{pre}, \mathsf{eff}\rangle$ of partial assignments to V, called preconditions and effects, and $cost : O \to \mathbb{R}^{0+}$ is an operator cost function. Applying operator o in state s results in a state denoted by $s[\![o]\!]$. By the transition graph $\mathcal{T}_\Pi = \langle S, E\rangle$ of Π we refer to the edge-labeled digraph induced by Π over S: if $o \in O$ is applicable in state s, then $\mathcal{T}_\Pi$ contains an edge $(s, s[\![o]\!]; o)$ from s to $s[\![o]\!]$, labeled with o. For a task $\Pi = \langle V, O, s_0, G, cost\rangle$ and state $s \in S$, task $\Pi(s)$ is obtained from Π by setting the initial state to be s. Auxiliary notation: for $k \in \mathbb{N}$, $i \in [k]$ stands for $i \in \{1, 2, \dots, k\}$.

1. Offline: Find an equivalence relation $\sim \leq \sim_{\Gamma_{S_*}}$.
2. When evaluating the state s with $s \sim s'$ for some previously evaluated state s', if $g(s) \geq g(s')$, prune s as if it were never generated. Otherwise set g, parent, and act of s' to those of s. If *WA**, reopen s'.
3. If a goal state s_* is reached, (i) extract a sequence $\pi = \langle(\varepsilon, s_0), (o_1, s_1), \dots, (o_m, s_m)\rangle$ of pairs of state and action, where $s_m = s_*$, by the standard backchaining from s_* along the parent relation, setting actions by the act relation, and (ii) return $\mathsf{trace\text{-}forward}(\pi)$.

Figure 1: *GBFS*/*WA** extension to Γ_{S_*} symmetry breaking

The LAMA Planning System

Richter and Westphal (2010) provide a detailed description of LAMA, and thus here we briefly describe only the components relevant to our presentation later on. Using *GBFS* and then *WA**, LAMA employs two heuristics, each inducing its sets of preferred operators: the delete-relaxation FF heuristic (Hoffmann and Nebel 2001) and the landmark heuristic. The latter is based on disjunctive landmarks of the planning task, that is, sets of variable assignments of which one must occur at some point.

Given a state s and a set L of Π's landmarks, possibly annotated with some orderings, the *landmark heuristic* estimate of s is set to the *number of landmarks* $L(s)$ *yet to be achieved from* s *onwards* (Richter, Helmert, and Westphal 2008). When forward search reaches s for the first time via a sequence of operators π, $L(s)$ is set to $L \setminus (\mathsf{A}(s,\pi) \setminus \mathsf{RA}(s,\pi))$, where $\mathsf{A}(s,\pi) \subseteq L$ and $\mathsf{RA}(s,\pi) \subseteq \mathsf{A}(s,\pi)$ are the sets of *accepted* and *required again* landmarks, respectively. A landmark is accepted if it occurs at some state along π; the set $\mathsf{A}(s,\pi)$ is memorized as state property $\mathsf{A}(s)$. An accepted landmark is required again if it does not hold in s and it is a direct precondition of some landmark which is not accepted. From this point on, each time s is reached via this or another operator sequence π', LAMA performs a "multi-path" revision of the landmarks accepted at s by updating $\mathsf{A}(s)$ to $\mathsf{A}(s) \cap \mathsf{A}(s,\pi)$, and then recomputing $L(s)$ as above (Karpas and Domshlak 2009).

A^* Symmetry Breaking with Γ_{S_*}

An *automorphism* of a transition graph $\mathcal{T}_\Pi = \langle S, E\rangle$ is a permutation σ of the vertices S such that $(s, s'; o) \in E$ iff, for some o' with $cost(o') = cost(o)$, $(\sigma(s), \sigma(s'); o') \in E$. Automorphisms are closed under composition, forming the *automorphism group* $Aut(\mathcal{T}_\Pi)$ of the graph. $\Gamma \leq \Gamma'$ denotes that Γ is a subgroup of Γ'. Each subgroup of automorphisms $\Gamma \leq Aut(\mathcal{T}_\Pi)$ induces an equivalence relation $\sim_\Gamma$ on states

S: $s \sim_\Gamma s'$ iff $\sigma(s) = s'$ for some $\sigma \in \Gamma$. For a state subset $S' \subseteq S$, the subgroup $\Gamma_{S'} = \{\sigma \in \Gamma \mid \forall s \in S' : \sigma(s) \in S'\} \leq \Gamma$ is the *stabilizer* of $S_1, \dots, S_k$ with respect to Γ. Finally, a set of automorphisms Σ is said to *generate* a group Γ if Γ is the fixpoint of iterative composition of the elements of Σ. Finding a generating set of $Aut(G)$ for a graph G is not known to be polynomial-time, but backtracking search techniques are surprisingly effective in finding generating sets for substantial subgroups of $Aut(G)$.

Pruning symmetries by reasoning about automorphisms of the search space has been adopted in model checking (Emerson and Sistla 2011), constraint satisfaction (Puget 1993), and planning (Rintanen 2003; Fox and Long 1999; 2002; Pochter, Zohar, and Rosenschein 2011; Domshlak, Katz, and Shleyfman 2012). Here we build upon the recent approach of Domshlak, Katz, and Shleyfman (2012) for exploiting state space symmetries in cost-optimal planning using A^*, referred to here for brevity as DKS.

At the focus of DKS is a property of plans and goal-stabilizing automorphisms Γ_{S_*}: Let Π be a planning task, $\Gamma \leq \Gamma_{S_*}$, and $(s_0, s_1, \dots, s_k)$, $(s_0, s'_1, \dots, s'_l)$ be a pair of plans for Π. If, for some $i \in [k]$ and $i < j \in [l]$, $s_i = \sigma(s'_j)$ for some $\sigma \in \Gamma$, then $(s_0, \dots, s_{i-1}, \sigma(s'_j), \dots, \sigma(s'_l))$ is also a plan for Π, shorter than $(s_0, s'_1, \dots, s'_l)$. Based on that, DKS extends A^* search as follows: No matter which of the two states s_i and s'_j as above is generated second, it is pruned from the search. However, if s_i is the state generated second, then s'_j ceases represent itself and starts representing its Γ_{S_*}-symmetric counterpart s_i. For that "role switching" of s'_j, the parent s_{i-1} of s_i "adopts" s'_j as a pseudo-child and the operator o such that $s_i = s_{i-1}[\![o]\!]$ is memorized. These "state adoptions" then should be taken into account at plan extraction; for the respective procedure, we refer the reader to Domshlak, Katz, and Shleyfman (2012).

As the transition graph $\mathcal{T}_\Pi$ is not given explicitly, automorphisms of $\mathcal{T}_\Pi$ must be inferred from the description of Π. Following Pochter, Zohar, and Rosenschein (2011), the implementation of DKS by Domshlak, Katz, and Shleyfman (2012) is restricted to certain "syntactic" automorphisms $\Gamma^{\text{pdg}}_{S_*} \leq \Gamma_{S_*}$, corresponding to automorphisms of a compact, node-colored *problem description graph (PDG)*. As it was first observed by Pochter, Zohar, and Rosenschein (2011), every automorphism of Π's PDG explicitly induces an automorphism of $\mathcal{T}_\Pi$, and the former can be searched for using off-the-shelf tools for discovery of automorphisms in explicit, colored graphs, such as BLISS (Junttila and Kaski 2007). In addition, this search can be easily restricted to PDG automorphisms to stabilizers of S_*, that is, $\Gamma^{\text{pdg}}_{S_*}$. Finally, since finding the precise equivalence relation $\sim_\Gamma$ induced by the discovered subgroup $\Gamma \leq \Gamma^{\text{pdg}}_{S_*} \leq \Gamma_{S_*} \leq Aut(\mathcal{T}_\Pi)$ is NP-hard (Luks 1993), it is approximated (with a loss of precision, but not of correctness) via an equivalence relation $\sim \leq \sim_\Gamma$, defined by a heuristic local search in S, with the generators of Γ defining state neighborhood, and state evaluation being based on a lexicographic ordering of S (Pochter, Zohar, and Rosenschein 2011). Note that $s \sim s'$ implies $s' = \sigma(s)$ for $\sigma \in \Gamma$, which is derived from the local search paths from s and s' to the (same) canonical state.

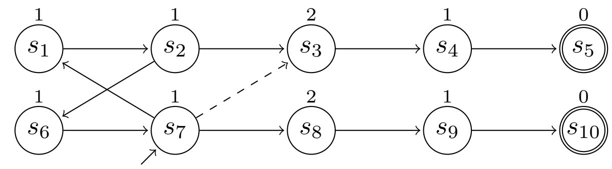

Figure 2: Illustration for Proposition 1

Satisfying Planning with Γ_{S_*}

Enhancing optimal A^* planning with DKS has been shown empirically effective, not only for reduction in expanded nodes, but also for increasing the overall coverage (Domshlak, Katz, and Shleyfman 2012). In principle, nothing prevents us from adopting DKS in satisficing planning; this is true whether the planning is based on *GBFS*, on *WA**, or on an iterative combination of the two as in LAMA. However, it is not clear whether the overhead of reasoning about symmetries pays off in satisficing planning, and if so, what the right way is to incorporate this reasoning into the search process. This question initiated our investigation, and in what follows, we discuss both our initial findings and some subsequent developments.

As a first step, we have implemented DKS within both *GBFS* and *WA** iterations of LAMA. The extension, described in Figure 1, is independent of the heuristic function, eagerness of the state evaluation, and both preferred operators and heuristic composition mechanisms. The only two differences from DKS in A^* are that (i) in *GBFS*, s' is not reopened in step 2, and (ii) to cover both lazy and eager heuristic evaluations, step 2 considers state s not when it is generated, but when it is about to be evaluated.

The potential benefit of adopting DKS in satisficing search is twofold. First, similarly to the effect obtained in A^*, no two states from the same equivalence class will ever be expanded. Second, the quality of the plans discovered with DKS is expected to be at least as good as, and possibly better than, the quality of the plans discovered without DKS. In particular:

Proposition 1 *Let $\sim \leq \sim_{\Gamma_{S_*}}$, and let h be a heuristic for a planning task Π that is invariant under $\sim$, i.e., $h(s) = h(s')$ holds for all $s \sim s'$. Then, assuming perfect tie-breaking, if π and π' are plans for Π found by WA* with and without reasoning about $\sim$, respectively, then $cost(\pi) \leq cost(\pi')$. Moreover, for any value of the WA* weight parameter, it is possible that $cost(\pi) < cost(\pi')$.*

The claim also holds for *GBFS* as the latter can be considered as *WA** for a sufficiently large weight. To see how DKS can actually improve the plans, consider a schematic example of a state space in Figure 2, where s_7 is the initial state, $S_* = \{s_5, s_{10}\}$, and the solid arcs depict the transitions. There are two plans: the longer plan to s_5, and the shorter one, to s_{10}. It is easy to see that, for $i \in [5]$, we have $s_i \sim s_{i+5}$. Assuming heuristic values as above the state nodes in Figure 2, *GBFS* with lazy evaluation may generate the longer plan. With DKS, however, *GBFS* in such a case will necessarily evaluate s_8 before evaluating s_4. State s_8 will then be found symmetric to the previously evaluated s_3, causing the initial state s_7 to "adopt" s_3 (dotted arc). At the end, when s_5 is reached, the plan extracted by trace-forward from the "plan to s_5" will actually be the shorter plan to s_{10}.

Landmark Heuristic and Γ_{S_*}

While the results show the pros of employing DKS in satisficing search, pruning symmetric states can also be detriment if the heuristics in use are not invariant under $\sim_{\Gamma_{S_*}}$, as is the case with both the FF and landmark heuristics used by LAMA. It is always possible that, on the states of some equivalence class, the heuristic is most inaccurate on the state that is evaluated first by the search procedure. Given that the rest of that equivalence class will be pruned, it is possible that the pruning takes the planning into a much longer search than what it would undergo without pruning. At a first view, a repair suggests itself almost immediately: in step 2, before discarding state s, compute $h(s)$ and use it to update $h(s')$. The difficulty with that repair is twofold. First, even if both $h(s)$ and $h(s')$ are computed, which of them should be used for s' is somewhat clear only if h is admissible, while most of the heuristics used to date in satisficing planning are inadmissible. Second, whether the heuristic evaluation is lazy or eager, computing heuristic values for pruned states eliminates part of the value that symmetry breaking brings to the search process in the first place.

We now show that, at least with the landmark heuristic, heuristic-related information between symmetric states can be communicated in a meaningful and cost-effective way. The basic idea corresponds to extending the multi-path inference of landmarks to "multi-state" inference between the symmetric states. Recall that each $\sigma \in \Gamma^{\text{pdg}}_{S_*}$ maps variable assignments to variable assignments. Let L be a set of disjunctive landmarks for Π, and let $\Gamma \leq \Gamma^{\text{pdg}}_{S_*}$. For each landmark $\varphi \in L$, and each $\sigma \in \Gamma$, by $\sigma(\varphi)$ we denote the set of variable assignments obtained by applying σ to each of the variable assignments in φ, that is, $\sigma(\varphi) = \{\langle\sigma(v), \sigma(d)\rangle \mid \langle v, d\rangle \in \varphi\}$. Similarly, by $\sigma(L')$ for $L' \subseteq L$, we denote the set $\{\sigma(\varphi) \mid \varphi \in L\}$.

Proposition 2 *Let Π be a planning task, $s, s' \in S$, and $\Gamma \leq \Gamma^{pdg}_{S_*}$. If $s' = \sigma(s)$ for some $\sigma \in \Gamma$, and φ is a landmark for $\Pi(s)$, then $\sigma(\varphi)$ is a landmark for $\Pi(s')$.*

The proof of Proposition 2 is almost immediate from the definitions of $\Gamma^{\text{pdg}}_{S_*}$ and landmarks. Note also that Proposition 2 is independent of how landmarks for different states of Π are discovered in the first place. In particular, landmarks for $\Pi(s)$ can either be restricted, as in LAMA, to the landmarks for $\Pi \equiv \Pi(s_0)$, or discovered specifically for $\Pi(s)$ (Helmert and Domshlak 2009; Bonet and Helmert 2010). In LAMA extended with DKS as in Figure 1, at step 2 we can update the set of landmarks $L(s')$ to be achieved from s onwards to $L(s') \cup \sigma(L(s))$. That is, if the search starts with a set L of landmarks for Π, then the sets of yet to be achieved landmarks $L(s)$ for states s of Π are no longer restricted to subsets of L, but to subsets of a (possibly much larger) set $\{\sigma(\varphi) \mid \varphi \in L, \sigma \in \Gamma^{\text{pdg}}_{S_*}\} \supseteq L$.

First, landmarks of state s reached by LAMA are computed at a very low effort from the landmarks of its parent state and the respective operator. Hence, the computational overhead of the multi-state inference of landmarks between the symmetric states remains low. Second, updating the heuristic estimate of the equivalence class representative s' this way results in a more accurate estimate of s', *subject to* validity of the assumption that knowing more landmarks of $\Pi(s')$ results in more accurate estimates of the goal distance from s'. This assumption does not hold for the landmark heuristic in general (or otherwise the latter would be admissible), but it is still the core assumption behind the landmark heuristic, similarly to how the "shorter relaxed plans are more accurate" assumption underlies the FF heuristic. Hence, multi-state inference of landmarks between the symmetric states is at least fully consistent with the concept of LAMA's landmark heuristic.

While Proposition 2 allows for inferring landmarks for $\Pi(s)$ that are not (or are not known to be) landmarks of Π, in our current extension of LAMA we restrict our inference to the initially discovered landmarks L of Π. The resulting modification of step 2 in Figure 1 is summarized by the following corollary of Proposition 2:

Corollary 3 Let L be a set of landmarks for a planning task Π, Γ be a subgroup of $\Gamma^{\text{pdg}}_{S_*}$, and $\varphi \in L$ be a landmark of Π. For any pair of states s and s', if $s' = \sigma(s)$ for some $\sigma \in \Gamma$, $\varphi \in L(s)$, and $\sigma(\varphi) \in L$, then $\sigma(\varphi) \in L(s')$.

Research Directions

At the next steps, we plan to examine the possibility of employing even more fine-grained symmetry exploitation in state space search. In principle, nothing should really bind us to only automorphisms of the searh space. The general family of state mappings that probably draws the boundary of symmetry breaking in state space search is this of distance-preserving homomorphisms, with automorphisms being its most studied and mathematically best grounded representative. Still, we believe that exploiting some non-automorphism mappings should be both possible and beneficial. One direction that we currently examine in this respect aims at connecting between state symmetries and some specific partial order reductions. Our departing point in this research is a special fragment of planning problems known as delete-free, or monotonic, planning (Bonet and Geffner 2001). This fragment is both interesting on its own (Riabov and Liu 2006; Gefen and Brafman 2011; Pommerening and Helmert 2012), as well as the most widely used basis for deriving heuristic estimates for general planning (Hoffmann and Nebel 2001; Helmert and Domshlak 2009). In the context of delete-free planning, we intend to expand our approach to symmetry pruning to (repeatedly) detecting state-symmetries during the search with dynamically excluded operators. If successful, we will work on generalizing the respective methodology to wider fragments of classical planning, hopefully making its beneficial even for general SAS^+ planning.

References

Bäckström, C., and Klein, I. 1991. Planning in polynomial time: The SAS-PUBS class. *Computational Intelligence* 7(3):181–197.

Bäckström, C., and Nebel, B. 1995. Complexity results for SAS^+ planning. *Computational Intelligence* 11(4):625–655.

Bonet, B., and Geffner, H. 1998. HSP: Heuristic search planner. In *AIPS'98 Planning Competition*.

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

Bonet, B., and Helmert, M. 2010. Strengthening landmark heuristics via hitting sets. In *Proceedings of the 19th European Conference on Artificial Intelligence*, 329–334.

Domshlak, C.; Katz, M.; and Shleyfman, A. 2012. Enhanced symmetry breaking in cost-optimal planning as forward search. In *Proceedings of the 22nd International Conference on Automated Planning and Scheduling (ICAPS)*.

Emerson, E. A., and Sistla, A. P. 2011. Symmetry and model checking. *Formal Methods in System Design* 9(1-2):105–131.

Fox, M., and Long, D. 1999. The detection and exploitation of symmetry in planning problems. In *Proceedings of the 16th International Joint Conference on Artificial Intelligence (IJCAI)*, 956–961.

Fox, M., and Long, D. 2002. Extending the exploitation of symmetries in planning. In *Proceedings of the 6th International Conference on Artificial Intelligence Planning Systems (ICAPS)*, 83–91.

Gefen, A., and Brafman, R. I. 2011. The minimal seed set problem. In *ICAPS*.

Haslum, P., and Geffner, H. 2000. Admissible heuristics for optimal planning. In *Proceedings of the 15th International Conference on Artificial Intelligence Planning Systems (AIPS)*, 140–149.

Helmert, M., and Domshlak, C. 2009. Landmarks, critical paths and abstractions: Whats the difference anyway? In *Proceedings of the 19th International Conference on Automated Planning and Scheduling (ICAPS)*.

Helmert, M., and Röger, G. 2008. How good is almost perfect? In *Proceedings of the 23rd AAAI Conference on Artificial Intelligence (AAAI)*, 944–949.

Helmert, M.; Haslum, P.; and Hoffmann, J. 2007. Flexible abstraction heuristics for optimal sequential planning. In *Proceedings of the 17th International Conference on Automated Planning and Scheduling (ICAPS)*, 200–207.

Helmert, M. 2004. A planning heuristic based on causal graph analysis. In *Proceedings of the Fourteenth International Conference on Automated Planning and Scheduling (ICAPS)*, 161–170.

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

Hoffmann, J., and Nebel, B. 2001. The FF planning system: Fast plan generation through heuristic search. *Journal of Artificial Intelligence Research* 14:253–302.

Junttila, T., and Kaski, P. 2007. Engineering an efficient canonical labeling tool for large and sparse graphs. In *ALENEX*, 135–149.

Karpas, E., and Domshlak, C. 2009. Cost-optimal planning with landmarks. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI-09)*.

Katz, M., and Domshlak, C. 2010. Implicit abstraction heuristics. *Journal of Artificial Intelligence Research* 39:51–126.

Luks, E. M. 1993. Permutation groups and polynomial-time computation. In *Groups and Computation, DIMACS Series in Disc. Math. and Th. Comp. Sci.*, volume 11. 139–175.

Pochter, N.; Zohar, A.; and Rosenschein, J. S. 2011. Exploiting problem symmetries in state-based planners. In *Proceedings of the 25th AAAI Conference on Artificial Intelligence (AAAI)*.

Pommerening, F., and Helmert, M. 2012. Optimal planning for delete-free tasks with incremental lm-cut. In *ICAPS*, *to appear*.

Puget, J.-F. 1993. On the satisfiability of symmetrical constrained satisfaction problems. In *Proceedings of the 7th International Symposium on Methodologies for Intelligent Systems (ISMIS)*, volume 689 of *LNCS*, 350–361.

Riabov, A., and Liu, Z. 2006. Scalable planning for distributed stream processing systems. In *ICAPS*, 31–41.

Richter, S., and Westphal, M. 2010. The LAMA planner: Guiding cost-based anytime planning with landmarks. *Journal of Artificial Intelligence Research* 39:127–177.

Richter, S.; Helmert, M.; and Westphal, M. 2008. Landmarks revisited. In *Proceedings of the 23rd AAAI Conference on Artificial Intelligence (AAAI-08)*, 975–982.

Richter, S.; Thayer, J. T.; and Ruml, W. 2010. The joy of forgetting: Faster anytime search via restarting. In *Proceedings of the 20th International Conference on Automated Planning and Scheduling (ICAPS)*, 137144.

Richter, S.; Westphal, M.; and Helmert, M. 2011. LAMA 2008 and 2011 (planner abstract). In *Seventh International Planning Competition (IPC 2011), Deterministic Part*. 50–54.

Rintanen, J. 2003. Symmetry reduction for SAT representations of transition systems. In *Proceedings of the 13th International Conference on Automated Planning and Scheduling (ICAPS)*, 32–41.