Stubborn Sets for Fully Observable Nondeterministic Planning

Dominik Winterer and **Yusra Alkhazraji**
University of Freiburg, Germany
{wintered, alkhazry}@cs.uni-freiburg.de

Michael Katz
IBM Watson Health, Israel
katzm@il.ibm.com

Martin Wehrle
University of Basel, Switzerland
martin.wehrle@unibas.ch

Abstract

Pruning techniques based on strong stubborn sets have recently shown their potential for SAS$^+$ planning as heuristic search. Strong stubborn sets exploit operator independency to safely prune the search space. Like SAS$^+$ planning, fully observable nondeterministic (FOND) planning faces the state explosion problem. However, it is unclear how stubborn set techniques carry over to the nondeterministic setting. In this paper, we introduce stubborn set pruning to FOND planning. We lift the notion of *strong* stubborn sets and introduce the conceptually more powerful notion of *weak* stubborn sets to FOND planning. Our experimental analysis shows that weak stubborn sets are beneficial to an LAO* search, and in particular show favorable performance when combined with symmetries and active operator pruning.

Introduction

During the execution of operator sequences, an agent may face situations where she has no control over certain aspects of the world. To handle such situations, it can be helpful to capture the nondeterminism of the world's dynamic within the domain model. Fully observable nondeterministic planning (FOND) formalism achieves that by allowing operators to have multiple outcomes. That is essentially its key difference from classical planning. Solutions to FOND planning tasks are policies that map states to actions. A policy determines the next action in the final plan, depending on the actual outcome of the previously applied action.

Like in classical planning, a key challenge in FOND planning is the state explosion problem, i.e., the exponential blowup of the state space induced by a compact description of planning tasks. In classical planning, and specifically in SAS$^+$ planning (Bäckström and Nebel 1995), the state explosion problem is often tackled by state pruning techniques (Alkhazraji et al. 2012; Wehrle et al. 2013; Wehrle and Helmert 2014; Domshlak, Katz, and Shleyfman 2012; Chen and Yao 2009). Out of these, strong stubborn sets are of a particular interest. Roughly speaking, strong stubborn sets can prune operators in a state if there is an alternative "permutation" plan starting in the same state, and the first operator in the alternative plan is preserved. For example, two operators that work on entirely different state variables can be permuted, and if both are applicable in a state s, both application orders in s will result in the same state. Strong stubborn sets will recognize this property of such independent operators, and only allow for applying one of them in s, hence pruning states in the induced transition system.

Although SAS$^+$ and FOND planning tasks are syntactically similar, differing only in whether operators may have multiple outcomes, the semantics of these formalisms is different. In particular, "classical" state spaces induced by SAS$^+$ tasks (OR graphs) differ from the more complex state spaces represented by AND/OR graphs induced by FOND planning tasks. This difference poses a challenge for carrying the state pruning techniques from classical planning over to fully observable nondeterministic planning. Recently, symmetry based techniques were adapted in a rather straight-forward way to FOND planning (Winterer, Wehrle, and Katz 2016). Strong stubborn sets, however, cannot be carried over to FOND planning in a straight-forward way. It is an open question how strong stubborn sets can be lifted to deal with AND/OR graphs, and in particular, how an accurate notion of operator interference for nondeterministic operators would look like.

In this paper we develop the theoretical basis for stubborn sets in FOND planning. We empirically evaluate these pruning techniques, exploiting stubborn sets for pruning LAO* search in AND/OR graphs. Further, we evaluate whether stubborn sets can contribute on top of other pruning techniques, such as symmetry elimination and active operators (Winterer, Wehrle, and Katz 2016; Wehrle et al. 2013). In more detail, the paper provides the following contributions.

- We generalize the theory of stubborn set pruning from SAS$^+$ planning to FOND planning. The generalization is based on a corresponding notion of interference.
- We introduce *weak* stubborn sets for FOND planning. Like strong stubborn sets, weak stubborn sets (Valmari 1989) have been originally introduced for model checking Petri nets (Petri 1962). Weak stubborn sets potentially allow for more pruning than strong stubborn sets, but require a more involved description for the original formalism used by Valmari (1989). Unlike strong stubborn sets, weak stubborn sets have not been investigated for planning before. We also introduce a nondeterministic variant of strong stubborn sets for FOND planning.

- We empirically evaluate the practical potential of stubborn set pruning to FOND planning with LAO*. In particular, inspired by recent work for classical planning that combines strong stubborn sets and symmetry elimination (Wehrle et al. 2015), we combine weak stubborn sets with structural symmetries for FOND tasks (Winterer, Wehrle, and Katz 2016) and with active operators pruning (Chen and Yao 2009; Wehrle et al. 2013). Our empirical investigation shows that the combination of all these pruning techniques yields promising performance gains on top of the respective baselines.

Preliminaries

We use an SAS$^+$ based notation to model fully observable nondeterministic planning tasks with a finite set of finite-domain state variables $\mathcal{V}$. Every variable v in $\mathcal{V}$ has a finite domain $dom(v)$. An assignment $v = d$ for $v \in \mathcal{V}$ and $d \in dom(v)$ is called a *fact*. A *partial state* s is a function from variables $vars(s) \subseteq \mathcal{V}$ to values in the domains of $vars(s)$, whereas all variables in $\mathcal{V} \setminus vars(s)$ have an undefined value $\bot$. We denote the value of v in s with $s[v]$ (including $s[v] = \bot$ if $v \in \mathcal{V} \setminus vars(s)$). A *state* is a partial state where all values are defined, i.e., with $vars(s) = \mathcal{V}$. We consider *fully observable nondeterministic* (FOND) planning tasks. As a key difference to classical planning, operators in FOND planning tasks can have several effects. Only one of these effects is applied when the operator is executed. Formally, FOND planning tasks are defined as follows.

Definition 1 (fully observable nondeterministic planning task)**.** *A fully observable nondeterministic (FOND) planning task is a tuple* $\Pi = \langle \mathcal{V}, \mathcal{O}, s_0, s_\star \rangle$*, where*

- $\mathcal{V}$ *is a finite set of finite-domain state variables,*
- $\mathcal{O}$ *is a finite set of nondeterministic operators,*
- s_0 *is the initial state,*
- $s_\star$ *is a partial state, called goal state.*

A *nondeterministic operator* $o \in \mathcal{O}$ has the form $\langle pre(o) \mid \mathit{effs}(o) \rangle$, where $pre(o)$ is a partial state that denotes the *precondition* of o and $\mathit{effs}(o)$ is the set of *effects* of o, where each effect $\mathit{eff}^{\,i}(o) \in \mathit{effs}(o)$ is a partial state. The degree of an operator o, denoted as $deg(o)$, is the number of its nondeterministic effects. Deterministic operators have degree of 1, i.e., $\mathit{effs}(o) = \{\mathit{eff}^{\,1}(o)\}$. For simplicity, we write $\mathit{eff}(o)$ instead of $\mathit{eff}^{\,1}(o)$ for deterministic operators. Throughout this paper, we assume unit costs for all operators.

In the following, we will denote a *nondeterministic operator* as an *operator* unless stated otherwise. The *all-outcome determinization* of nondeterministic operator o is $o^{[1]}, \dots, o^{[k]}$ ($k = deg(o)$) where $o^{[i]} = \langle pre(o), \{\mathit{eff}^{\,i}(o)\} \rangle$ for $1 \leq i \leq k$ is called an *outcome* of o. We use $\mathcal{O}_{det}$ to denote the set of all outcomes of operators in $\mathcal{O}$. An operator o is *applicable* in a state s iff $s[v] = pre(o)[v]$ holds for all $v \in vars(pre(o))$. If o has an empty precondition (i.e., it has no precondition), then o is applicable in every state. The empty precondition is denoted by $\top$. If operator o is applicable in s, the set of *successor states* $o(s) := \{o^{[i]}(s) \mid i \in \{1, \dots, deg(o)\}\}$ of s is obtained from s by applying the outcomes $o^{[i]}$ in s, where the state $o^{[i]}(s)$ is obtained from s by setting the values of variables in $vars(\mathit{eff}^{\,i}(o))$ to their values in $\mathit{eff}^{\,i}(o)$ and leaving the remaining variable values unchanged. For simplicity of notation: If $o(s)$ includes a single state, we refer to it as a state rather than a set of states. We denote the set of applicable operators in s by $app(s)$.

Solutions to FOND planning tasks are policies subsuming the plan notion of plans in classical planning (Cimatti et al. 2003). Formally, a *policy* is a mapping $\pi : S \mapsto \mathcal{O} \cup \{\bot\}$, which maps states to operators or π is undefined (i.e. $\pi(s) = \bot$). We denote the sequential application of the operators determined by π as *following* π. A policy π is called *weak* if π defines at least one path from the initial state to a goal state when following π. In this case, π is called a *weak plan* for Π. A policy π is *closed* if following π either leads to a goal state, or to a state where the policy is defined. It is *proper* if from every state visited following π, there exists a path to a goal state following π. A policy that is closed and proper is called a *strong cyclic plan* for Π, the plan notion of interest thorough this paper. Furthermore, π is *acyclic* if it does not revisit already visited states. A closed and proper acyclic policy is called a *strong plan* for Π.

Informally, a weak plan is a sequence of operators which leads to the goal if all nondeterministic operators' outcomes were deterministic, which corresponds to the plan notion in classical planning. A strong plan guarantees that a goal state is reached, and an upper bound on the number of plan steps exists. In contrast to strong plans, strong cyclic plans reach a goal state after a number of steps, but no such upper bound on the plan steps can be provided a priori. However, strong cyclic planning is based on the fairness assumption, i.e., there is a nonzero probability that a goal state can be reached when following a strong cyclic plan.

Stubborn Sets for Classical Planning

The stubborn set method was originally introduced in computer-aided verification and used for dead end detection in Petri nets (Valmari 1989). Stubborn sets have been investigated and evaluated for optimal planning in the work of Alkhazraji et al. (2012). In a nutshell, a stubborn set for some state s is a sufficient subset of operators such that restricting the successor generation to the applicable operators within this stubborn set preserves a permutation of each plan from s. In this section, we will introduce all necessary preliminaries for stubborn sets in classical planning.

Definition 2 (disabling, conflicting, strong and weak interference)**.** *Let* o_1 *and* o_2 *be deterministic operators.*

- o_1 disables o_2 *if there exists some* $v \in vars(\mathit{eff}(o_1)) \cap vars(pre(o_2))$ *with* $\mathit{eff}(o_1)[v] \neq pre(o_2)[v]$.
- o_1 *and* o_2 conflict *if there exists some* $v \in vars(\mathit{eff}(o_1)) \cap vars(\mathit{eff}(o_2))$ *with* $\mathit{eff}(o_1)[v] \neq \mathit{eff}(o_2)[v]$.
- o_1 *and* o_2 strongly interfere *if* o_1 *disables* o_2*,* o_2 *disables* o_1*, or* o_1 *and* o_2 *conflict.*
- o_1 weakly interferes *with* o_2 *if* o_1 *disables* o_2*, or* o_1 *and* o_2 *conflict.*

In contrast to strong interference, weak interference is not a symmetric relation. In other words, if operator o_1 weakly interferes with operator o_2, it does *not* imply that o_2 weakly interferes with o_1. The two interference definitions are approximations in the sense that they are state-independent whereas the true operator dependencies depend on the current state. There are two further ingredients for the stubborn sets. Let s be a state. A *disjunctive action landmark* for a partial state s' in s is a set of operators with the property that at least one operator from this set occurs in every operator sequence that leads from s to any state that satisfies s'. A *necessary enabling set* for an operator o in s is a disjunctive action landmark for $pre(o)$ in s.

Two variants of stubborn sets are known in the literature: *strong* and *weak* stubborn sets. Their sole difference is the interference relation. In planning, however, only the strong variant of stubborn sets has been investigated so far. In our work, we will also investigate weak stubborn sets. Based on Definition 2, the notion of weak stubborn sets for classical planning boils down to the following definition.

Definition 3 (weak stubborn sets)**.** *Let $\Pi = \langle \mathcal{V}, \mathcal{O}, s_0, s_\star \rangle$ be a classical planning task and s be a state. A set $T_s \subseteq \mathcal{O}$ is a weak stubborn set (WSS) in s if:*

1. *T_s contains a disjunctive action landmark for $s_\star$ in s.*
2. *For each operator $o \in T_s$ with $o \notin app(s)$, T_s contains a necessary enabling set for o in s.*
3. *For each operator $o \in T_s$ with $o \in app(s)$, T_s contains all operators o' with which o weakly interferes.*

If we exchange weak interference by strong interference, we end up with a *strong* stubborn set. Obviously, weak stubborn sets dominate strong stubborn sets in terms of pruning power, in the sense that every strong stubborn set is a weak stubborn set, but not vice versa. The proof of retaining completeness and optimality given by Alkhazraji et al. (2012) directly generalizes to weak interference. We emphasize that there is a difference in weak and strong interference computation. While strong interference might be refined by mutex reasoning (i.e., two operators o_1 and o_2 cannot strongly interfere if there is a variable $v \in vars(pre(o_1)) \cap vars(pre(o_2))$ and $pre(o_1)[v] \neq pre(o_1)[v]$), the mutex reasoning violates the completness of weak stubborn sets. We refer to our technical report for details.

Stubborn Sets for FOND

For classical planning tasks, the solutions are consecutive operator applications leading from the initial state to one of the goal states. Hence, for an operator sequence $o_1, \dots, o_n$ of pairwise independent operators, all permutations of $o_1, \dots, o_n$ lead to the same state. This property is the core idea of stubborn sets. For FOND planning tasks, however, solutions cannot be represented as consecutive operator sequences and it is unclear how strong cyclic plans permute. As a first idea to face this theoretical problem, we might determinize the nondeterministic operators and use the stubborn set theory from classical planning. More precisely: For a given nondeterministic planning problem Π, a straightforward approach would be to directly apply the original definition of strong stubborn sets on the all-outcome determinization of Π, and additionally, to add for every outcome $o^{[i]}$ of a nondeterministic operator o every other outcome of o to the candidate stubborn set, in order to respect o's nondeterministic nature. However, as the following example shows, such an approach is incomplete.

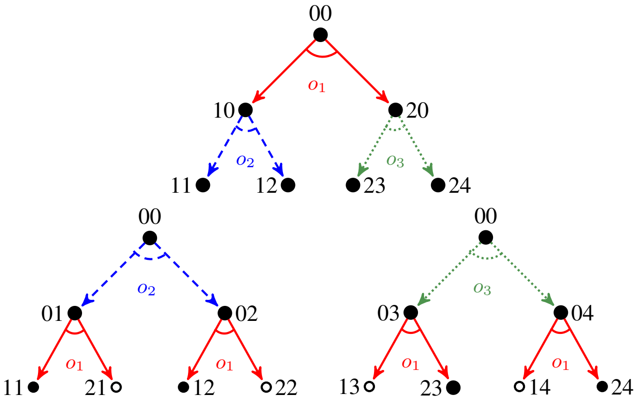


Figure 1: Postponing o_1 in s_0 only leads to policies with dead-ends. The straightforward instantiation is incomplete.

Example 1. *Consider the following all-outcome determinization $\Pi_{det} = \langle \mathcal{V}, \mathcal{O}_{det}, s_0, s_* \rangle$ of nondeterministic planning task Π with variables $\mathcal{V} = \{v_1, v_2\}$ and the following operators:*

- $o_1^{[1]} = \langle v_1 = 0 \mid \{v_1 := 1\} \rangle$, $o_1^{[2]} = \langle v_1 = 0 \mid \{v_1 := 2\} \rangle$
- $o_2^{[1]} = \langle v_2 = 0 \mid \{v_2 := 1\} \rangle$, $o_2^{[2]} = \langle v_2 = 0 \mid \{v_2 := 2\} \rangle$
- $o_3^{[1]} = \langle v_2 = 0 \mid \{v_2 := 3\} \rangle$, $o_3^{[2]} = \langle v_2 = 0 \mid \{v_2 := 4\} \rangle$
- $o_{11} = \langle v_1 = 1, v_2 = 1 \mid \{v_2 := 5\} \rangle$
- $o_{12} = \langle v_1 = 1, v_2 = 2 \mid \{v_2 := 5\} \rangle$
- $o_{23} = \langle v_1 = 2, v_2 = 3 \mid \{v_2 := 5\} \rangle$
- $o_{24} = \langle v_1 = 2, v_2 = 4 \mid \{v_2 := 5\} \rangle$

The initial state is $s_0 = \{v_1 = 0, v_2 = 0\}$, and the goal is $s_\star = \{v_2 = 5\}$. The set $\{o_{11}, o_{12}, o_{23}, o_{24}\}$ is a disjunctive action landmark in s_0 which we add to the candidate stubborn set T_{s_0}. As all operators in this set are inapplicable in s_0, we have to add a necessary enabling set for all of them. A valid choice for these necessary enabling sets is based on selecting the unsatisfied conditions $v_2 = 1$, $v_2 = 2$, $v_2 = 3$ and $v_2 = 4$ in the preconditions of $o_{11}, o_{12}, o_{23}, o_{24}$, respectively, and to add the determinized operators that set these conditions to true. These achieving operators correspond to all outcomes of o_2 and o_3, which are applicable in s_0 but not weakly interfering with any operator not in T_{s_0}. Hence, we finally get $T_{s_0} = \{o_{11}, o_{12}, o_{23}, o_{24}, o_2^{[1]}, o_2^{[2]}, o_3^{[1]}, o_3^{[2]}\}$. However, T_{s_0} is insufficient for our purpose because every strong plan from s_0 has to start with o_1: Depending on the nondeterministic outcome of o_1 ($v_1 := 1$ or $v_1 := 2$), o_2 or o_3 can be applied to satisfy the precondition of an operator to reach the goal. In contrast, starting with o_2 and applying o_1 afterwards might lead to outcomes where no goal is reachable any more (e.g., $v_1 = 2$ and $v_2 = 2$). The analogous situation occurs when starting with o_3 and applying o_1 afterwards (Figure 1).

We observe: Weak interference in the all-outcome determinization is an insufficient criterion for the coupling behaviour of nondeterministic operators and the stubborn sets cannot be carried over to FOND planning in a straightforward way. Hence, other ideas are needed. We follow a stepwise approach. First, we give a new criterion based on deterministic operators. Then we prove a syntactic version of the criterion. Finally, we adapt the criterion to the FOND setting and use it as a basis for our pruning technique.

We introduce the idea of an operator being *attached* to the front of a weak plan in a given state such that the resulting sequence remains a weak plan from that state (Figure 2). Formally, the following definition reflects this idea.

Definition 4 (attachable operator). *Let o_1 and o_2 be operators in $\mathcal{O}_{det}$, π be an operator sequence, and s be a state. Operator o_1 is attachable to o_2 in state s if for every weak plan $o_2\pi$ from s, $o_1 o_2 \pi$ is a weak plan from s.*

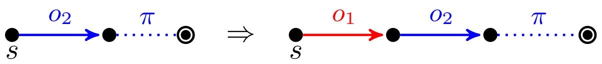


Figure 2: Adding o_1 to the front of the operator transforms weak plan $o_2\pi$ into weak plan $o_1 o_2 \pi$.

Checking the attachability property is computationally intractable. To be precise, this property is state-dependent, i.e., it needs to be checked in each visited state. Moreover, verifying this property amounts to finding two weak plans for s, one that starts with o_2 and another one that starts with o_1. Therefore, it is necessary to come up with an approximation based on a syntactic check of operators which we formulate using two new definitions. Let $\Pi = \langle \mathcal{V}, \mathcal{O}, s_0, s_\star \rangle$ be a FOND planning task and o_1 be an operator in $\mathcal{O}_{det}$. The *disabled set* $dis(o_1)$ is defined as the set of operator-variable pairs $(o_2, v) \in \mathcal{O}_{det} \times \mathcal{V}$, such that o_1 disables o_2 on v, i.e., $v \in vars(eff(o_1)) \cap vars(pre(o_2))$ and $eff(o_1)[v] \neq pre(o_2)[v]$. The *negated goals set* $neg(o_1)$ is defined as the set of goal variables with which o_1 conflicts, i.e., $eff(o_1)[v] \neq s_\star[v]$.

Definition 5 (syntactic attachable operator). *Let o_1 and o_2 be two operators in $\mathcal{O}_{det}$. We say that o_1 is* syntactically attachable *to o_2 if*

1. *o_1 does not disable o_2,*
2. *$dis(o_1) \subseteq dis(o_2)$, and*
3. *$neg(o_1) \subseteq neg(o_2)$.*

The syntactic attachability property is state-independent and does not suffice on its own to verify attachability since it states nothing about the applicability of operators. For instance, o_1 can be inapplicable in state s in which o_2 is applicable, while all the conditions in Definition 5 hold. Therefore, stipulating that o_1 is applicable in s, in addition to being syntactically attachable to o_2, is essential for implying attachability.

Theorem 1. *Let $\Pi = \langle \mathcal{V}, \mathcal{O}, s_0, s_\star \rangle$ be a FOND planning task, s be a state of Π, and o_1 and o_2 be operators in $\mathcal{O}_{det} \cap app(s)$. If o_1 is syntactically attachable to o_2, then o_1 is attachable to o_2 in s.*

For the proof of Theorem 1 we again refer to the technical report. In the following section we emphasize the difference between weak interference and attachability.

Weak Interference vs. Attachability

According to weak interference, we obtain equivalent weak plans by permuting weakly non-interfering operators. On the other hand, attachability allows introducing new operators to the beginning of weak plans such that the resulting sequence is a weak plan, too. As a first step towards a stubborn set formalism for FOND planning, we clarify the relationship between attachability and weak interference. At first sight, it might seem that attachability subsumes weak interference. However, as it turns out, both notions are incomparable, i.e., if operator o_1 is syntactically attachable to operator o_2, this does not imply that o_1 does not weakly interfere with o_2.

Example 2. *Consider a FOND planning task Π with variables $\mathcal{V} = \{v_1, v_2, v_3, v_4\}$ and the following operators:*

- $o_1 = \langle v_1 = 0 \mid \{v_2 := 1, v_4 := 1\} \rangle$
- $o_2 = \langle v_1 = 0 \mid \{v_3 := 1, v_4 := 2\} \rangle$

The initial state is $s_0 = \{v_1 = 0, v_2 = 0, v_3 = 0, v_4 = 0\}$, the goal partial state $s_\star = \{v_2 = 1, v_3 = 1\}$. We observe that operators o_1 and o_2 are mutually syntactically attachable, i.e., $dis(o_1) \subseteq dis(o_2) = \emptyset$, and $neg(o_1) \subseteq neg(o_2) = \emptyset$. However, they also mutually weakly interfere because they conflict on variable v_4. This fundamental difference of weak interference and attachability gives rise to a new concept of state space pruning. We obtain a stubborn set based solely on attachability by replacing rule 3 in Definition 3 with the following rule:

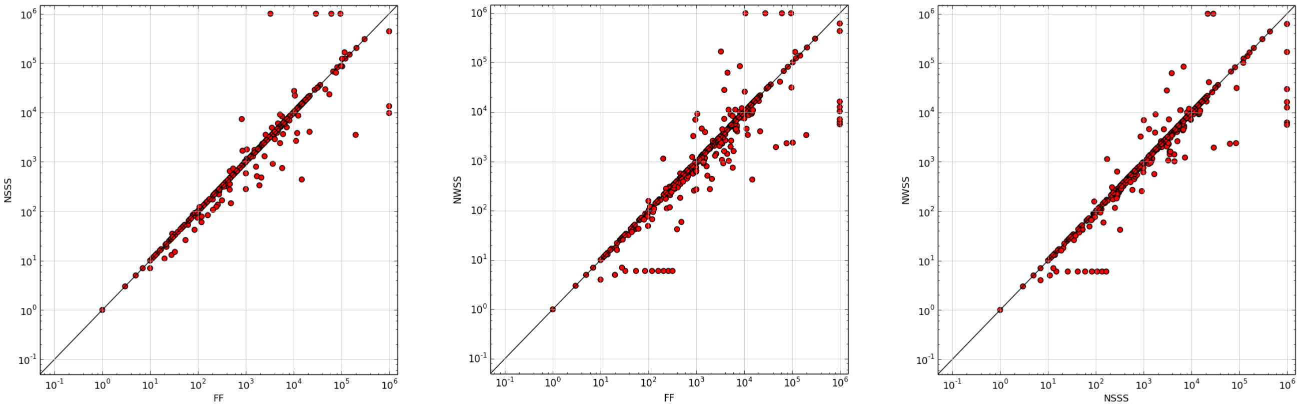
3′. *For each operator $o \in T_s$ with $o \in app(s)$, T_s contains all operators o' to which o is not syntactically attachable.*

This yields a new notion of attachability-based stubborn sets which is a completeness-preserving pruning technique [1]. However, due to the nature of attachability, it can lead to suboptimal weak plans. We further show, by the following example, that stubborn sets based on weak interference and stubborn sets based on attachability are two different pruning techniques such that none of them dominates the other with respect to pruning power.

Example 3. *Consider a FOND planning task Π with variables $\mathcal{V} = \{v_1, v_2, v_3\}$ and the following operators:*

- $o_1 = \langle \top \mid \{v_1 := 2, v_2 := 1\} \rangle$
- $o_2 = \langle \top \mid \{v_1 := 1\} \rangle$

[1] We do not provide a formal proof, as it is an intermediate step towards our contribution. However, it is easy to see that attachability is a sufficient condition for completeness: $dis(o_1) \subseteq dis(o_2)$ guarantees that v_4 does not occur in some operator's precondition s.t. o_2 enables that operator w.r.t. v_4 and o_1 disables it. Hence, conflicting on v_4 is not critical in this case for preserving plans.



(a) (b) (c)

Figure 3: Generated nodes comparison for (a) *NSSS* vs. plain *FF*, (b) *NWSS* vs. plain *FF*, and (c) *NWSS* vs. *NSSS*.

- $o_3 = \langle v_1 = 1, v_2 = 1 \mid \{v_3 := 1\}\rangle$

The initial state is $s_0 = \{v_1 = 0, v_2 = 0, v_3 = 0\}$ and the goal state is $s_\star = \{v_1 = 1, v_2 = 1, v_3 = 1\}$. Let T_a denote a stubborn set that uses only syntactic attachability. Since $\{o_2\}$ is a disjunctive action landmark for $s_\star$ in s_0, we can use it to initialize the stubborn set: $T_a(s_0) = \{o_2\}$. Because $o_2 \in app(s_0)$, we need to add the operators to which o_2 is not attachable. Since $dis(o_2) = \emptyset$ and $neg(o_2) = \emptyset$, o_2 is attachable to both o_1 and o_3. This means that the computation will terminate on $T_a(s_0) = \{o_2\}$. On the other hand, if we compute T_s, the stubborn set based on weak interference, the result is different. We initialize $T_s(s_0)$ with o_2 as before. o_2 is applicable, which means we need to add operators with which o_2 weakly interferes. Obviously, o_2 weakly interferes with o_1 because they conflict on variable v_1. Therefore, o_1 is added to $T_s(s_0)$. Also, o_1 is applicable and it disables o_3, hence o_3 is added to $T_s(s_0)$, which means all operators are applied in s_0 ($T_s(s_0) = \{o_1, o_2, o_3\}$).

Furthermore, each method can result in a different plan: depending on the search algorithm, using only syntactic attachability might result in plan $o_2 o_1 o_2 o_3$, while using only weak interference might lead to $o_1 o_2 o_3$, which is optimal.

Example 4. *Consider a FOND planning task Π with variables $\mathcal{V} = \{v_1, v_2\}$ and the following operators:*

- $o_1 = \langle \top \mid \{v_1 := 1\}\rangle$
- $o_2 = \langle \top \mid \{v_2 := 1\}\rangle$
- $o_3 = \langle v_1 = 0 \mid \{v_2 := 1\}\rangle$

The initial state is $s_0 = \{v_1 = 0, v_2 = 0\}$ and the goal state $s_\star = \{v_1 = 1, v_2 = 1\}$. We compute $T_a(s_0)$: The set $\{o_1\}$ is a disjunctive action landmark in s_0, so we add it to $T_a(s_0)$. Since o_1 is applicable, we add operators to which it is not attachable. Because $dis(o_1) = \{o_3\} \nsubseteq dis(o_2) = \emptyset$, o_1 is not attachable to o_2 and therefore o_2 is added to $T_a(s_0)$. In addition, o_3 is added because o_1 disables o_3 and hence not attachable to it, ending in $T_a(s_0) = \{o_1, o_2, o_3\}$. Using weak interference: we add o_1 to $T_s(s_0)$ as a disjunctive action landmark. It is applicable so we need to add the operators with which o_1 weakly interferes. We add o_3 only because o_1 disables o_3, but we do not add o_2 because neither o_1 nor o_3 weakly interferes with o_2. Therefore, $T_s(s_0) = \{o_1, o_3\}$. We conclude this section with the following corollary, summarizing our observations.

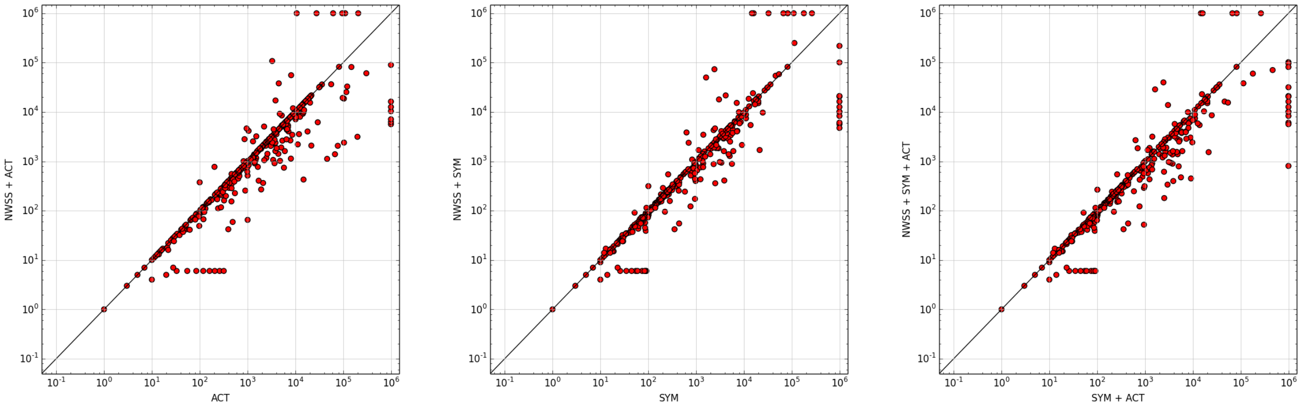
Corollary 1. *Stubborn sets based on weak interference and stubborn sets based on attachability are incomparable in terms of pruning power.*

Nondeterministic Weak Stubborn Sets

While attachability is a sufficient criterion for stubborn sets pruning, it could lead to unnecessary state explorations (e.g. see plan length in Example 3). We propose a stubborn set method based on a combination of attachability and weak interference for FOND planning. The definition of a disjunctive action landmark can be extended to FOND planning in a straightforward way. In the context of FOND planning, a *disjunctive action landmark* for a partial state s' in a state s is a set of nondeterministic operators with the property that some outcome of an operator from this set occurs on every path from s to any state that satisfies s'. A *necessary enabling set* for an operator o in s is a disjunctive action landmark for $pre(o)$ in s. The following definitions extend syntactic attachability and weak interference to nondeterministic operators. Let o_1 and o_2 be two operators in $\mathcal{O}$. We say that o_1 *accords* with o_2 if outcome $o_1^{[i]}$ is syntactically attachable to outcome $o_2^{[j]}$ for all $i \in \{1, \dots, deg(o_1)\}$ and all $j \in \{1, \dots, deg(o_2)\}$. Let o_1 and o_2 be two operators in $\mathcal{O}$. We say that o_1 *weakly interferes* with o_2 if some outcome $o_1^{[i]}$ weakly interferes with some $o_2^{[j]}$, with $i \in \{1, \dots, deg(o_1)\}$ and $j \in \{1, \dots, deg(o_2)\}$. Based on these definitions we define weak stubborn sets for FOND planning.

Definition 6 (nondeterministic weak stubborn set)**.** *Let $\Pi = \langle \mathcal{V}, \mathcal{O}, s_0, s_\star\rangle$ be a nondeterministic planning task, and s be a state. A set $T_s \subseteq \mathcal{O}$, is a nondeterministic weak stubborn set (NWSS) in s if the following conditions hold:*

1. *T_s contains a disjunctive action landmark for $s_\star$ in s.*
2. *For each operator $o \in T_s$ with $o \notin app(s)$, T_s contains a necessary enabling set for o in s.*



(a) (b) (c)

Figure 4: Generated nodes comparisons for *NWSS* on top of other pruning techniques (a) active operators, (b) symmetries, and (c) both active operators and symmetries.

3. *For each operator $o \in T_s$ with $o \in app(s)$, T_s contains all nondeterministic operators o' ($deg(o') > 1$) with which o does not accord.*
4. *For each operator $o \in T_s$ with $o \in app(s)$, T_s contains all operators with which o weakly interferes.*

Nondeterministic weak stubborn sets can be seen as hybrid notion combining attachability with weak interference. Points (1) and (2) of Definition 6 are the same in Definition 3. Accordance, in point (3), is a necessary property for changing the application order of nondeterministic operators in AND/OR graphs. However, in some cases, pruning can be performed by merely permuting operator sequence (e.g. if the involved operators are deterministic) without the need to attach extra operators, which cannot be guaranteed by accordance alone. For this reason, we have chosen to make the distinction between deterministic and operators with degree at least two and use *only* weak interference when processing deterministic operators (point 4), i.e., if a planning task has only deterministic operators, then only points (1), (2), and (4) of Definition are relevant. In the following theorem we state the main result of this paper.

Theorem 2. *Restricting the successor generation to an NWSS in every state is completeness-preserving for strong cyclic planning.*

Proof. Let s be a state from which a strong cyclic plan exists and let π be one such plan. Let T_s be a *nondeterministic weak stubborn set* for s. We show that either (i) the operator $\pi(s)$ is in T_s, or (ii) there exists $o \in T_s$ such that $o \in app(s)$ and there exists a strong cyclic plan from each $s' \in o(s)$.

Let $\pi' = o_1 \dots o_n$ be any sequence of operators from π that defines an acyclic weak plan from s. T_s contains a disjunctive action landmark for $s_\star$ in s and thus it contains an operator from π'. Let o_i be such an operator with smallest index. Then $o_i \in app(s)$ (otherwise its necessary enabling set would be contained in T_s, mandating an operator from the necessary enabling set to appear on π' before o_i). If $i = 1$, we are done. Otherwise, consider two cases. If the operator o_i is deterministic, we see that $o_i^{[j]} o_1 \dots o_{i-1} o_{i+1} o_n$ is a weak plan from s and thus a strong cyclic plan. If o_i is a strictly nondeterministic operator, for every outcome $o_i^{[j]}$, $o_i^{[j]}\pi'$ is a weak plan as $o_i^{[j]}$ accords with every operator of smaller index within π'. We thus obtain a strong cyclic plan by first applying o_i in s, followed by applying π' in every possible outcome in $o_i(s)$. □

We also provide a nondeterministic *strong* variant of stubborn set (*NSSS*) by modifying Definition 6 as follows: In point 3, all operators that are do not *mutually* accord with o are added to the stubborn set. In point 4, all operators that *strongly* interfere with o are added to the stubborn set.

Corollary 2. *NSSS inherit the completeness property from NWSS.*

Proof. The definition of *NSSS* subsumes the definition of *NWSS*; hence, the proof of Theorem 2 shows also that *NSSS* is a completeness-preserving pruning technique. □

Experimental Evaluation

In this section, we empirically investigate the potential of nondeterministic stubborn sets when applied as the only pruning technique or on top of other pruning techniques such as structural symmetries and active operators pruning. For our investigation, we have implemented both strong and weak stubborn set variants, as well as active operators on top of an adaptation of Fast Downward (Helmert 2006) to FOND planning (Winterer, Wehrle, and Katz 2016), which already included symmetry based pruning. Our code is available upon request.

As baseline for our investigations, we employ LAO* using FF heuristic (Hoffmann and Nebel 2001) (based on all-outcome determinization of FOND planning tasks). All configurations are built on top of this baseline. For all stubborn sets approaches, the disabling relation and achievers were entirely precomputed, while the interference relation was computed during the search and then cached for later use. The benchmark set consists of all IPC-08 FOND domains, scaled-up versions of these domains by Christian Muise and

Coverage	FF	$NSSS$	$NWSS$	PRP
BLOCKSWORLD (30)	22	21	21	**30**
CHAIN-OF-ROOMS-FIXED (10)	**10**	**10**	**10**	**10**
EARTH-OBSERVATION (40)	33	33	33	**40**
FAULTS (55)	**55**	**55**	**55**	**55**
FIRST-RESPONDERS (100)	98	97	98	**100**
FOREST (90)	8	10	13	**75**
BLOCKSWORLD-NEW (50)	16	15	15	**37**
ELEVATORS (15)	14	14	**15**	**15**
EX-BLOCKSWORLD (15)	8	8	8	**9**
FAULTS-NEW (190)	**190**	**190**	**190**	**190**
FIRST-RESPONDERS-NEW (95)	30	30	31	**94**
FOREST-NEW (90)	7	7	8	**88**
TIDYUP-MDP (10)	**10**	**10**	**10**	**10**
TIREWORLD (15)	**15**	**15**	**15**	**15**
TRIANGLE-TIREWORLD (40)	7	7	7	**40**
Sum (845)	523	522	529	**808**

Table 1: Coverage of baseline (FF), strong stubborn sets ($NSSS$) and weak stubborn sets ($NWSS$) and the state of the art FOND planner (PRP).

other FOND domains commonly used in the literature. All our experiments were conducted on a cluster equipped with Intel Xeon E5-2650 v2 CPUs running at 2.6 GHz . For each run, the time limit and memory bound were 30 minutes and 2 GB respectively.

Stubborn Sets for FOND

First of all, we want to find out whether nondeterministic strong and nondeterministic weak stubborn sets lead to noticeable performance improvement. To that end, we compare the configurations applying pruning with each of the stubborn sets on top of the baseline (FF). The per domain coverage is depicted in Table 1 and per instance generated nodes are shown in Figure 3, pairwise comparing the three configurations. The configurations applying nondeterministic strong and weak stubborn sets are denoted by $NSSS$ and $NWSS$, respectively. First, note that $NSSS$ performs similarly to FF, loosing one task each in three domains, and increasing coverage by two tasks in one domain. In total, coverage is reduced by one instance which goes along with only slight reductions in node generations: In Figure 3a, most of the tasks appear on or near the diagonal.

Comparing $NWSS$ to FF, it looses one task each in two domains and gains five tasks in one domain and one task each in three domains, overall increasing coverage by six tasks. Looking at Figure 3b, indeed pruning helps in most cases to reduce the number of generated nodes, in a number of cases the decrease almost reaching two orders of magnitude. Note that there is an exponential reduction in generated nodes on a part of the FIRST-RESPONDERS domain. Figure 3c shows that this gain carries over to the comparison with $NSSS$, reflecting the theoretical dominance of nondeterministic weak stubborn sets over the strong variant. It should be pointed out that due to the non-optimal nature of the LAO* algorithm, the theoretical dominance does not necessarily translate to a dominance in the number of generated nodes. To summarize, nondeterministic weak stubborn are beneficial to an LAO* search whereas nondeterministic strong stubborn sets are benefical to a lesser extent.

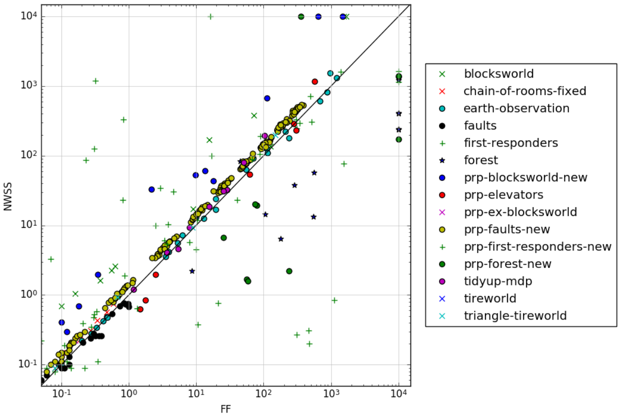


Figure 5: Total time for $NWSS$ vs. the baseline (FF).

It is worthwhile to mention, in domains where no pruning is possible, the stubborn sets computation can slow down the search on certain instances. We observe such an effect in the domains BLOCKSWORLD and PRP-BLOCKSWORLD-NEW where less instances are solved by stubborn sets than without (see Figure 5). On the other hand, as IPC planners using stubborn sets have demonstrated such coverage losses, can be easily avoided (e.g. by stopping the stubborn sets computation after a predefined threshold of unsuccessfully pruning attempts is exceeded).

Combination with other Pruning Techniques

Recently, it has been shown that structural symmetries and stubborn sets can be successfully combined for classical planning (Wehrle et al. 2015). Hence, a natural question arises: Do we get similar results when combining weak stubborn sets with structural symmetries for FOND planning? And more generally, to which extent can we observe synergy effects when combining pruning techniques for FOND planning? To take a step towards these questions, we investigate combinations of nondeterministic weak stubborn sets with the following two pruning techniques.

- *Symmetry Elimination*: Symmetry elimination considers equivalence classes of symmetrical states and allows for using representative states of each equivalence class. Many approaches have shown their potential in several contexts in classical planning. We use the variant of structural symmetries for FOND planning, recently proposed by Winterer, Wehrle, and Katz (2016).
- *Active operators pruning*: The pruning technique was introduced by Chen and Yao (2009) and further investigated by Wehrle et al. (2013). In a nutshell, given a state s, an operator o is considered *active* if there exists a weak plan from s starting with o. A sufficient criterion can be formulated based on domain transition graph. The successor generation in a state s is restricted to active operators.

We remark that combining these pruning techniques in a straightforward fashion results in a combined pruning tech-

Pruning Technique	**Coverage**	
	all ops	*active ops*
FF	523	524
$NWSS$	529	527
SYM	526	527
$NWSS + SYM$	532	**538**

Table 2: Coverage results for all combinations of $NWSS$, SYM and active operator pruning (*all ops* vs *active ops*).

.

nique that is safe. For our investigation, we consider all possible combinations of symmetry elimination and active operators pruning with and without nondeterministic weak stubborn sets. We report the overall coverage results in Table 2. Going beyond the overall coverage, Figure 4 depicts the generated nodes comparison. We observe that all non-combined pruning techniques improve coverage. Further, nondeterministic weak stubborn sets is the most beneficial configuration out of those (529 solved instances), followed by symmetry elimination (526 solved instances) and active operator pruning (524 solved instances). Especially, the configuration with additional active operator pruning leads to a substantial coverage increase of 15 additionally solved instances (compared to the baseline FF) which reflects the fact that this configuration is also clearly generating the least nodes (see Figure 4). As another insight, we see that applying $NWSS$ on top of symmetry reduction (SYM) leads to a clear performance increase, both with and without additional active operators pruning. Evaluating the gain from applying active operators pruning, this pruning technique is only moderately beneficial both as a single pruning technique and when combined with symmetry reduction and nondeterministic stubborn sets.

To investigate the anytime performance of our approaches, we provide a plot in Figure 6, reflecting for each time point the number of tasks solved by each of the configurations up to that time. Observe that the combination of all three methods dominates all other configuration. Also it can be seen that combining nondeterministic weak stubborn sets with symmetry elimination improves the runtime behavior, which makes them worth using in a practical setting.

Related Work

Pruning techniques in classical planning are directly related to this work. Beside stubborn sets, there is a variety of partial order reduction methods that have been applied to planning (Coles and Coles 2010; Nissim, Apsel, and Brafman 2012; Chen and Yao 2009). Moreover, symmetries have been investigated for classical planning as well (Fox and Long 1999; 2002; Rintanen 2003; Coles and Coles 2010; Pochter, Zohar, and Rosenschein 2011; Domshlak, Katz, and Shleyfman 2012; 2013; Shleyfman et al. 2015).

Our work is also related to various approaches in FOND planning. The LAO*algorithm was also used for FOND planning by Mattmüller et al. (2010) and was originally

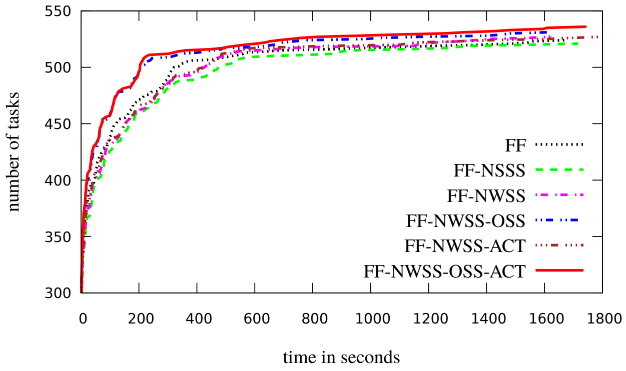


Figure 6: Number of planning tasks solved until timeout.

introduced in the context of MDPs (Hansen and Zilberstein 2001). As a key difference to the LAO*algorithm, determinization-based approaches work on the all-outcome determinization of FOND planning tasks. The current state of the art is represented by *determinization-based* planner *Planner for Relevant Policy* (PRP) by Muise, McIlraith, and Beck (2012). PRP uses multiple runs of a classical planner and regression to find a strong cyclic plan for a given FOND problem. Another recent regression-based FOND planner is *GRENDEL* (Ramírez and Sardiña 2014). Unlike PRP, GRENDEL does not rely on determinization but combines regression with a symbolic fixed-point computation. In addition, approaches that generalize classical planning concepts to FOND planning are particularly related to our work. For example, conditional effects for FOND planning (Muise, McIlraith, and Belle 2014), or pattern databases (Mattmüller et al. 2010).

Conclusion

We provided the theoretical basis for stubborn sets pruning in FOND planning by introducing two stubborn set variants for FOND: nondeterministic weak and strong stubborn sets. We empirically showed that nondeterministic weak stubborn sets are beneficial to an LAO* search. Moreover, we showed that the dominance of nondeterministic weak stubborn sets over the strong variant carries over in practice. We also investigated the combination of nondeterministic weak stubborn sets with symmetry elimination and active operators pruning. The empirical results show that combining all three pruning methods leads to the strongest configuration. For future work, encouraged by this insight, it is particularly interesting to investigate the potential of combining other pruning techniques in the context of classical and FOND planning. Moreover, as pruning techniques may slow down the search on certain instances, it is beneficial to design algorithms that can detect (prior to search) whether or not a particular pruning technique should be used during search.

Acknowledgements

We thank the anonymous reviewers for their keen eyes and helpful suggestions. Dominik Winterer wants to thank

Robert Mattmüller for helpful discussions while supervising his bachelor thesis that served as a basis for this work.

References

Alkhazraji, Y.; Wehrle, M.; Mattmüller, R.; and Helmert, M. 2012. A stubborn set algorithm for optimal planning. In *Proc. ECAI 2012*, 891–892.

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

Chen, Y., and Yao, G. 2009. Completeness and optimality preserving reduction for planning. In *Proc. IJCAI 2009*, 1659–1664.

Cimatti, A.; Pistore, M.; Roveri, M.; and Traverso, P. 2003. Weak, strong, and strong cyclic planning via symbolic model checking. *Artif. Intell.* 147(1-2):35–84.

Coles, A. J., and Coles, A. 2010. Completeness-preserving pruning for optimal planning. In *Proc. ECAI 2010*, 965–966.

Domshlak, C.; Katz, M.; and Shleyfman, A. 2012. Enhanced symmetry breaking in cost-optimal planning as forward search. In *Proc. ICAPS 2012*.

Domshlak, C.; Katz, M.; and Shleyfman, A. 2013. Symmetry breaking: Satisficing planning and landmark heuristics. In *Proc. ICAPS 2013*, 298–302.

Fox, M., and Long, D. 1999. The detection and exploitation of symmetry in planning problems. In *Proc. IJCAI 1999*, 956–961.

Fox, M., and Long, D. 2002. Extending the exploitation of symmetries in planning. In *Proc. AIPS 2002*, 83–91.

Hansen, E. A., and Zilberstein, S. 2001. LAO*: A heuristic search algorithm that finds solutions with loops. *AIJ* 129(1–2):35–62.

Helmert, M. 2006. The Fast Downward planning system. *JAIR* 26:191–246.

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

Mattmüller, R.; Ortlieb, M.; Helmert, M.; and Bercher, P. 2010. Pattern database heuristics for fully observable nondeterministic planning. In *Proc. ICAPS 2010*, 105–112.

Muise, C. J.; McIlraith, S. A.; and Beck, J. C. 2012. Improved non-deterministic planning by exploiting state relevance. In *Proc. ICAPS 2012*, 172–180.

Muise, C. J.; McIlraith, S. A.; and Belle, V. 2014. Nondeterministic planning with conditional effects. In *Proc. ICAPS 2014*.

Nissim, R.; Apsel, U.; and Brafman, R. I. 2012. Tunneling and decomposition-based state reduction for optimal planning. In *Proc. ECAI 2012*, 624–629.

Petri, C. A. 1962. *Kommunikation mit Automaten*. Ph.D. Dissertation, Darmstadt University of Technology, Germany.

Pochter, N.; Zohar, A.; and Rosenschein, J. S. 2011. Exploiting problem symmetries in state-based planners. In *Proc. AAAI 2011*, 1004–1009.

Ramírez, M., and Sardiña, S. 2014. Directed fixed-point regression-based planning for non-deterministic domains. In *Proc. ICAPS 2014*.

Rintanen, J. 2003. Symmetry reduction for SAT representations of transition systems. In *Proc. ICAPS 2003*, 32–40.

Shleyfman, A.; Katz, M.; Helmert, M.; Sievers, S.; and Wehrle, M. 2015. Heuristics and symmetries in classical planning. In *Proc. AAAI 2015*, 3371–3377.

Valmari, A. 1989. Stubborn sets for reduced state space generation. In *Proc. APN 1989*, 491–515.

Wehrle, M., and Helmert, M. 2014. Efficient stubborn sets: Generalized algorithms and selection strategies. In *Proc. ICAPS 2014*, 323–331.

Wehrle, M.; Helmert, M.; Alkhazraji, Y.; and Mattmüller, R. 2013. The relative pruning power of strong stubborn sets and expansion core. In *Proc. ICAPS 2013*, 251–259.

Wehrle, M.; Helmert, M.; Shleyfman, A.; and Katz, M. 2015. Integrating partial order reduction and symmetry elimination for cost-optimal classical planning. In *Proc. IJCAI 2015*, 1712–1718.

Winterer, D.; Wehrle, M.; and Katz, M. 2016. Structural symmetries for fully observable nondeterministic planning. In *Proc. IJCAI 2016*, 3293–3299.