Proceedings of the Thirty-Fourth International Conference on Automated Planning and Scheduling (ICAPS 2024)

On the Computational Complexity of Stackelberg Planning and Meta-Operator Verification

Gregor Behnke[1], **Marcel Steinmetz**[2]
[1]ILLC, University of Amsterdam, Amsterdam, The Netherlands
[2]LAAS-CNRS, Université de Toulouse, France
g.behnke@uva.nl, marcel.steinmetz@laas.fr

Abstract

Stackelberg planning is a recently introduced single-turn two-player adversarial planning model, where two players are acting in a joint classical planning task, the objective of the first player being hampering the second player from achieving its goal. This places the Stackelberg planning problem somewhere between classical planning and general combinatorial two-player games. But, where exactly? All investigations of Stackelberg planning so far focused on practical aspects. We close this gap by conducting the first theoretical complexity analysis of Stackelberg planning. We show that in general Stackelberg planning is actually no harder than classical planning. Under a polynomial plan-length restriction, however, Stackelberg planning is a level higher up in the polynomial complexity hierarchy, suggesting that compilations into classical planning come with a worst-case exponential plan-length increase. In attempts to identify tractable fragments, we further study its complexity under various planning task restrictions, showing that Stackelberg planning remains intractable where classical planning is not. We finally inspect the complexity of meta-operator verification, a problem that has been recently connected to Stackelberg planning.

Introduction

Stackelberg planning (Speicher et al. 2018a) is an adversarial planning problem, in which two agents/players act consecutively in a joint classical planning task. The objective of the first player (called the *leader*) is to choose and to play a plan that maximally raises the cost of the second player (the *follower*) to subsequently achieve its goal. This type of planning is useful for real-world adversarial settings commonly found in the cyber-security domain (Speicher et al. 2018b; Di Tizio et al. 2023). To solve Stackelberg planning tasks, there so far exists just a single generic algorithm paradigm called *leader-follower search* (Speicher et al. 2018a), which searches over possible leader plans, solving a classical planning task for each. As the number of possible plans is exponential in the worst case, this makes one wonder how the complexity of Stackelberg planning relates to classical planning. Focusing on algorithmic improvements (Speicher et al. 2018a; Torralba et al. 2021; Sauer et al. 2023), existing works have neglected this question so far.

We close this gap, providing the first theoretical analysis of Stackelberg planning's complexity. Stackelberg planning is a special case of general combinatorial two-player games (Stockmeyer and Chandra 1979). And, indeed, as many combinatorial games can, Stackelberg planning is reducible to fully-observable non-deterministic (FOND) planning (Cimatti, Roveri, and Traverso 1998), using action effect non-determinism to emulate all possible follower choices. This narrows down Stackelberg planning's complexity to the range between classical planning (Bylander 1994) and FOND planning (Littman 1997). We show that Stackelberg planning is PSPACE-complete, and thus is in fact not harder than classical planning in general. However, Stackelberg planning is Σ_2^{P}-complete under a polynomial plan-length restriction. This relates to results in FOND (Rintanen 1999) and conformant planning (Baral, Kreinovich, and Trejo 2000), and contrasts the NP-completeness of the corresponding classical planning problem (Jonsson and Bäckström 1998). Hence, unless $\mathrm{NP} = \Sigma_2^{\mathrm{P}}$, polynomial compilations of Stackelberg planning into classical planning have a worst-case exponential plan-length blow-up.

The analysis of tractable fragments has shown to be an important source for the development of domain-independent heuristic in classical planning (e.g., Hoffmann and Nebel 2001; Domshlak, Hoffmann, and Katz 2015). With the vision of establishing a basis for the development of leader-follower search heuristics, we analyze the complexity of Stackelberg planning under various syntactic restrictions. An overview of our results is given in Tab. 1.

Lastly, we explore a problem related to Stackelberg planning: *meta-operator* (Pham and Torralba 2023) verification. Meta-operators are action-sequence wild cards, which can be instantiated freely for every state satisfying the operator's precondition as long as operator's effects match. Pham and Torralba have cast verifying whether a given action is a valid meta-operator as a Stackelberg planning task. We show that meta-operator verification PSPACE-complete and Π_2^{P}-complete under a polynomial plan-length restriction.

Proofs are provided in a technical report (Behnke and Steinmetz 2024).

Background

Classical Planning We assume STRIPS notation (Fikes and Nilsson 1971). A planning task is a tuple $\Pi =$

Syntactic restrictions	Plan existence		Optimal planning		METAOPVER
	PLANSAT	STACKELSAT	PLANMIN	STACKELMIN	
$*$ preconds $*$ effects $\|\pi\|$ not bounded	PSPACE	PSPACE (Theorem 1)	PSPACE	PSPACE (Theorem 2)	PSPACE (Theorem 10)
$*$ preconds $*$ effects $\|\pi\| \in \mathcal{O}(n^k)$	NP	Σ_2^{p} (Theorem 3)	NP	Σ_2^{p} (Theorem 3)	Π_2^{p} (Theorem 11)
1 precond 1+ effect	NP	Σ_2^{p} (Theorem 4)	NP	Σ_2^{p} (Corollary 1)	–
$*$+ preconds 1 effect	P	NP (Theorem 5)	NP	Σ_2^{p} (Theorem 7)	–
0 preconds 2 effects	P	P for ∞ effects (Theorem 6)	NP	Σ_2^{p} (Theorem 8)	–
0 preconds 1 effect non-unit cost	P	P for ∞ effects (Theorem 6)	P	NP (Theorem 9)	–

Table 1: Overview of our complexity results. For comparison, the PLANSAT and PLANMIN columns show the complexity of classical planning under the respective task restrictions, as given by (Bylander 1994). All results prove completeness with respect to the different complexity classes. $*$ means arbitrary number, $+$ only positive, $*+$ arbitrary positive, and $n+$ n positive.

$\langle V, A, I, G\rangle$ consisting of a set of propositional *state variables* (or *facts*) V, a set of *actions* A, an *initial state* $I \subseteq V$, and a *goal* $G \subseteq V$. For $p \in V$, p and $\neg p$ are called *literals*. A *state* s is a subset of V, with the interpretation that all state variables not in s do not hold in s. Each action $a \in A$ has a *precondition* $pre(a)$, a conjunction of literals, an *add effect* (also called positive effect) $add(a) \subseteq V$, a *delete effect* (negative effect) $del(a) \subseteq V$, and a non-negative *cost* $c(a) \in \mathbb{N}_0$. A planning task has *unit costs* iff for all actions $c(a) = 1$. a is applicable in a state s iff $s \models pre(a)$. Executing a in s yields the state $s[\![a]\!] = (s \setminus del(a)) \cup add(a)$. These definitions are extended to action sequences π in an iterative manner. The cost of π is the sum of costs of its actions. π is called an *s-plan* if π is applicable in s and $G \subseteq s[\![\pi]\!]$. π is an *optimal* s-plan if $c(\pi)$ is minimal among all s-plans. An (optimal) plan for Π is an (optimal) I-plan. If there is no I-plan, we say that Π is *unsolvable*. Two decision problem formulations of classical planning are considered in the literature. *PLANSAT* is the problem of given a planning task Π, deciding whether there exists any plan for Π. *PLANMIN* asks, given in addition a (binary-encoded) cost bound B, whether there is a plan π for Π with cost $c(\pi) \leq B$. Both problems are known to be PSPACE-complete (Bylander 1994).

Stackelberg Planning A Stackelberg planning task (Speicher et al. 2018a) is a tuple $\Pi^{LF} = \langle V, A^L, A^F, I, G^F\rangle$, where the set of actions is partitioned into one for each player. A *leader plan* is an action sequence $\pi^L = \langle a_1^L, \dots, a_n^L\rangle \in (A^L)^n$ that is applicable in I. π^L induces the *follower task* $\Pi^F(\pi^L) = \langle V, A^F, I[\![\pi^L]\!], G^F\rangle$. An (optimal) *follower response* to π^L is an (optimal) plan for $\Pi^F(\pi^L)$. We denote by $c^F(\pi^L)$ the cost of the optimal follower response to π^L, defining $c^F(\pi^L) = \infty$ if $\Pi^F(\pi^L)$ is unsolvable. Leader plans are compared via a dominance order between cost pairs where $\langle c_1^L, c_1^F\rangle$ *weakly dominates* $\langle c_2^L, c_2^F\rangle$ ($\langle c_1^L, c_1^F\rangle \sqsubseteq \langle c_2^L, c_2^F\rangle$), if $c_1^L \leq c_2^L$ and $c_1^F \geq c_2^F$. $\langle c_1^L, c_1^F\rangle$ (strictly) *dominates* $\langle c_2^L, c_2^F\rangle$ ($\langle c_1^L, c_1^F\rangle \sqsubset \langle c_2^L, c_2^F\rangle$), if $\langle c_1^L, c_1^F\rangle \sqsubseteq \langle c_2^L, c_2^F\rangle$ and $\langle c_1^L, c_1^F\rangle \neq \langle c_2^L, c_2^F\rangle$. To simplify notation, we write $\pi_1^L \sqsubset \pi_2^L$ if $\langle c(\pi_1^L), c^F(\pi_1^L)\rangle \sqsubset \langle c(\pi_2^L), c^F(\pi_2^L)\rangle$. A leader plan π^L is optimal if it is not dominated by any leader plan. Previous works have considered algorithms for computing the set of all optimal solutions, called the *Pareto frontier*.

Stackelberg Planning Decision Problems

We distinguish between two decision-theoretic formulations of Stackelberg planning, akin to classical planning:

Definition 1 (STACKELSAT)**.** *Given Π^{LF}, STACKELSAT is the problem of deciding whether there is a leader plan π^L that makes $\Pi^F(\pi^L)$ unsolvable.*

Definition 2 (STACKELMIN)**.** *Given Π^{LF}, and two binary-encoded numbers $B^L, B^F \in \mathbb{N}_0$. STACKELMIN is the problem of deciding whether there is a leader plan π^L with $\langle c(\pi^L), c^F(\pi^L)\rangle \sqsubseteq \langle B^L, B^F\rangle$.*

Interpreting the leader's objective as rendering the follower's objective infeasible, the first definition directly mirrors the PLANSAT plan-existence decision problem. Similarly, the second definition mirrors PLANMIN in looking for solutions matching a given quantitative cost bound. It is worth mentioning that both decision problems are implicitly looking for only a single point in the Pareto frontier, whereas previous practical works dealt with algorithms computing this frontier entirely. In terms of computational complexity, this difference is however unimportant. In particular, answering even just a single STACKELMIN question does in fact subsume the computation of the entire Pareto frontier – if the answer is no, one necessarily had to compare the given bounds to *every* element in the Pareto frontier.

As in classical planning, STACKELSAT can be easily (with polynomial overhead) reduced to STACKELMIN:

Proposition 1. *STACKELSAT is polynomially reducible to STACKELMIN.*

Given that Stackelberg planning is a proper generalization of classical planning, the Stackelberg decision problems are guaranteed to be at least as hard as the respective classical planning decision problem. By applying Immerman–Szelepcsényi theorem (Szelepcsényi 1987; Immerman 1988), we can prove that it is also no harder than classical planning in the general case:

Theorem 1. *STACKELSAT is PSPACE-complete.*

Theorem 2. *STACKELMIN is PSPACE-complete.*

In spite of these results, algorithms for Stackelberg planning are significantly more complicated than their classical planning counterparts. In particular, the results raise the question of whether it is possible to leverage the classical planning methods directly for solving Stackelberg tasks via compilation. Polynomial compilations necessarily exist as per the theorems, yet, it is interesting to investigate which "side-effects" these might need to have. In order to investigate these questions, we turn to a more fine granular analysis by considering the complexity under various previously studied syntactic classes of planning tasks.

Stackelberg Planning under Restrictions

Polynomial Plan Length

For classical planning, it is commonly known that restricting the length of the plans to be *polynomial* in the size of the planning task description, makes the decision problems become NP-complete (Jonsson and Bäckström 1998).

Definition 3 (Polynomial Stackelberg Decision)**.** *Given Π^{LF} with non-0 action costs, and two binary-encoded numbers $B^L, B^F \in \mathbb{N}_0$ that are bounded by some polynomial $p \in \mathcal{O}(\ell^k)$ for $\ell = |V| + |A^L| + |A^F|$. STACKELPOLY is the problem of deciding whether there is a leader plan π^L such that $\langle c(\pi^L), c^F(\pi^L)\rangle \sqsubseteq \langle B^L, B^F\rangle$.*

We restrict the action cost to be strictly positive, ensuring that considering leader and follower plans with polynomial length is sufficient to answer the decision problem. STACKELPOLY is harder than the corresponding classical problem.

Theorem 3. *STACKELPOLY is Σ_2^p-complete.*

This result implies that, unless NP $= \Sigma_2^{\mathrm{p}}$ which would collapse the polynomial hierarchy (Arora and Barak 2007, Theorem 5.6), polynomial compilations of Stackelberg planning into classical planning come with a worst-case exponential plan-length increase.

Bylander's Syntactic Restrictions

Bylander (1994) studied the complexity of classical planning under various syntactic restrictions, drawing a concise borderline between planning's tractability and infeasibility. Bylander distinguishes between different planning task classes based on the number of action preconditions and effects, and the existence of negative preconditions or effects. Table 1 provides an overview of the main classes. Here, we take up his analysis and show that even for the classes where classical planning is tractable, Stackelberg may not be. We consider STACKELSAT and STACKELMIN in this order.

Definition 4. *Let $m, n \in \mathbb{N}_0 \cup \{\infty\}$. STACKELSAT$_n^m$ is the problem of deciding STACKELSAT for Stackelberg tasks so that $|pre(a)| \leq m$ and $|add(a)| + |del(a)| \leq n$ hold for all actions a. If m is preceded by "$+$", actions have no negative preconditions. If n is preceded by "$+$", actions have no delete effects. STACKELMIN$_n^m$ is defined similarly.*

We omit m (n) if $m = \infty$ ($n = \infty$). We consider only cases where the classical-planning decision problems are in NP. Stackelberg planning is PSPACE-hard when classical planning is.

Plan Existence

Bylander (1994) has shown that PLANSAT is already NP-complete for tasks with actions that even have just a single precondition and a single effect. Here we show that the corresponding Stackelberg decision problem is even one step above in the polynomial hierarchy:

Theorem 4. *STACKELSAT$_{+1}^1$ is Σ_2^p-complete.*

Bylander (1994) has shown that PLANSAT is polynomial if only positive preconditions and only a single effect per action are allowed. Even under these restrictive conditions, STACKELSAT however still remains intractable:

Theorem 5. *STACKELSAT$_1^+$ is NP-complete.*

Stackelberg plan-existence however becomes easy, when forbidding preconditions throughout. While this class of tasks seems to be trivial at first glance, optimal Stackelberg planning actually remains intractable as we show below.

Theorem 6. *STACKELSAT^0 is polynomial.*

Optimal Planning

As per Proposition 1, optimal Stackelberg planning is in general at least as hard as deciding plan existence. All intractability results shown for STACKELSAT carry over to STACKELMIN. As in all classes analyzed in the previous section, the consideration of polynomially length-bounded plans is sufficient for hardness, Σ_2^{p} yields a sharp upper bound to the complexity of STACKELMIN, per Theorem 3.

Corrolary 1. *STACKELMIN$_{+1}^1$ is Σ_2^p-complete.*

The results for STACKELSAT only provide a lower bound to the complexity of STACKELMIN. This lower bound may be strict as demonstrated by Thm. 7 and 8:

Theorem 7. *STACKELMIN$_1^{+1}$ is Σ_2^p-complete.*

Theorem 8. *STACKELMIN$_2^0$ is Σ_2^p-complete.*

Optimal Stackelberg planning remains intractable even when all actions have no preconditions and may have only at most one effect.

Theorem 9. *STACKELMIN$_1^0$ is NP-complete in general, but polynomial when additionally assuming unit cost.*

Complexity of Meta Operator Verification

Pham and Torralba (2023) have recently leveraged Stackelberg planning for synthesizing *meta-operators* in classical planning. Meta-operators, like macro-actions (Fikes and Nilsson 1971), are artificial actions that aggregate the effect of action sequences, therewith introducing shortcuts in state-space search. Formally, given a classical planning task Π and an action σ that is not in Π's action set, σ is a *meta-operator for Π* if, for every state $s \models pre(\sigma)$ that is reachable from I, there exists a sequence π of Π's actions such that $s[\![\sigma]\!] = s[\![\pi]\!]$. Whether a given σ is a meta-operator can be *verified* by solving a Stackelberg planning task.

Here, we consider the question whether using an expressive and computationally difficult formalism like Stackelberg planning is actually necessary. For this, we determine the computational complexity of meta-operator synthesis and compare it to that of Stackelberg planning.

Definition 5 (Meta-Operator Verification)**.** *Given Π and a fresh action σ. METAOPVER is the problem of deciding whether σ is a meta-operator for Π.*

Like for Stackelberg planning, the complexity of meta-operator verification in general remains the same as that of classical planning:

Theorem 10. *METAOPVER is PSPACE-complete.*

In other words, meta-operator verification could as well be compiled directly into a classical rather than a Stackelberg planning task. But how difficult or effective would such a compilation be? To shed light on this question, we again turn to a length bounded version of the problem.

Definition 6 (Polynomial Meta-Operator Verification)**.** *Given Π with non-*0 *action costs, a fresh action σ, and two binary-encoded numbers $B^P, B^M \in \mathbb{N}_0$ that are bounded by some polynomial $p \in \mathcal{O}(\ell^k)$ for $\ell = |V| + |A|$. polyMETAOPVER is the problem of deciding whether for all states $s \models pre(\sigma)$ reachable from I with a cost of at most B^P, there exists π with $c(\pi) \leq B^M$ and $s[\![\pi]\!] = s[\![\sigma]\!]$.*

The parameters B^P and B^M define the perimeter around the initial state respectively the reached state under which the meta-operator conditions are to be verified. As for Stackelberg planning, we require that the cost of all actions is strictly positive, which together with the cost bounds ensures that the radius of the perimeter is polynomially bounded.

Polynomial meta-operator verification too is on the second level of the polynomial hierarchy. Again, this means that under the assumption that the polynomial hierarchy does not collapse, polynomial compilations of meta-operator verification into classical planning in the worst case, come with an exponential plan-length blow-up.

Theorem 11. *polyMETAOPVER is Π_2^P-complete.*

Note that polyMETAOPVER is therefore in the co-complexity-class of polynomial Stackelberg plan-existence, i.e., they belong to co-classes on the same level of the polynomial hierarchy. This may not be surprising given the subtle difference between meta-operator verification and Stackelberg plan existence: while the latter asks for the existence of a (leader) action sequence where all induced (follower) action sequences satisfy some property, meta-operator verification swaps the quantifiers.

We want to point out that this duality can be exploited further, showing analogous results for Bylander's (1994) task classes. Contrary to Stackelberg planning, however, the identification of tractable fragments is less useful for meta-operator verification due to the lack of the monotonicity invariance of the meta-operator condition. An action being a meta-operator in a task abstraction does not imply that the action is a meta-operator in the original task, and vice versa. We hence do not further explore this analysis here.

Related Work

Stackelberg planning is strongly linked to conditional planning under partial observability (Bonet and Geffner 2000) and non-deterministic actions (Cimatti, Roveri, and Traverso 1998). Conditional plan existence in partially observable deterministic (POD) planning can be seen as the co-problem to STACKELSAT for the Stackelberg task where the leader enumerates the initial states, and the follower has the POD task's actions and goal. As hinted at in the introduction, there is a simple reduction of Stackelberg plan existence to FOND plan existence. In a nutshell, the FOND planning task inherits the leader actions and contains a single action with a non-deterministic effect per original follower action, and which non-deterministically sets a termination flag $\mathtt{T}$ (initially false). Leader planning and follower planning are split into two separate phases, distinguished by auxiliary facts. In the leader phase, it is possible to apply only leader actions and an action that will cause a transition into the follower phase. In the follower phase, only the follower's non-deterministic action can be applied. There is a Stackelberg plan iff there is a strong cyclic plan for the goal $\neg G^F \wedge \mathtt{T}$.

Both planning under partial observability and planning under non-deterministic effects are more challenging problems than Stackelberg planning in the general case, e.g., FOND planning is EXP-complete (Littman 1997) and POND planning even 2-EXP-complete (Rintanen 2004). But, there are interesting relations to our results under a polynomial plan length bound. Rintanen (1999) showed that polynomially-length-bounded conditional planning is Π_2^{P} complete, the co-result to our Thm. 3. His hardness proof is very similar to ours, with technical differences owed to the different planning formalisms. Bonet (2010) studied conditional planning with non-deterministic actions, proving that polynomially bounded plan existence for conditional plans with at most k branching points is $\Sigma_{2k+k}^{\mathrm{P}}$-complete. Stackelberg planning corresponds to $k = 1$, the difference between determinism and non-determinism causing the Σ_2^{P} vs. Σ_4^{P} complexity results. For conditional planning under partial observability, Baral, Kreinovich, and Trejo (2000) showed that plan existence is Σ_2^{P}-complete. Turner (2002) analyzed a wide range of different planning formalisms under a polynomial plan-length bound, but his formalism supported arbitrary boolean formulae as preconditions, making even length-1 plan existence already NP-complete.

Conclusion

Stackelberg planning remains PSPACE-complete, like classical planning, in general, but is Σ_2^P complete under a polynomial plan-length bound. Hence, unless the polynomial hierarchy collapses to its first level, a polynomial reduction of Stackelberg planning into classical planning is not possible in general. We showed that Stackelberg planning remains intractable under various syntactical restrictions, even in cases where classical planning is known to be tractable. Lastly, we have proven similar results for meta-operator verification, specifically PSPACE-completeness in general and Π_2^P-complete for the polynomial plan-length bounded case, with similar implications as the results for Stackelberg planning.

Acknowledgments

We thank the anonymous reviewers, whose comments helped to improve the paper. This work has received funding from the European Union's Horizon Europe Research and Innovation program under the grant agreement TUPLES No 101070149. Marcel Steinmetz would like to acknowledge the support of the Artificial and Natural Intelligence Toulouse Institute (ANITI), funded by the French Investing for the Future PIA3 program under the Grant agreement ANR-19-PI3A-000.

References

Arora, S.; and Barak, B. 2007. *Computational Complexity: A Modern Approach*. Princeton, Online Draft Version.

Baral, C.; Kreinovich, V.; and Trejo, R. 2000. Computational complexity of planning and approximate planning in the presence of incompleteness. *Artificial Intelligence*, 122(1-2): 241–267.

Behnke, G.; and Steinmetz, M. 2024. On the Computational Complexity of Stackelberg Planning and Meta-Operator Verification: Technical Report. https://arxiv.org/abs/2403.17826.

Bonet, B. 2010. Conformant plans and beyond: Principles and complexity. *Artificial Intelligence*, 174(3-4): 245–269.

Bonet, B.; and Geffner, H. 2000. Planning with Incomplete Information as Heuristic Search in Belief Space. In *Proceedings of the Fifth International Conference on Artificial Intelligence Planning Systems, AIPS 2000*, 52–61. AAAI.

Bylander, T. 1994. The Computational Complexity of Propositional STRIPS Planning. *Artificial Intelligence*, 69(1): 165–204.

Cimatti, A.; Roveri, M.; and Traverso, P. 1998. Strong Planning in Non-Deterministic Domains Via Model Checking. In *Proceedings of the Fourth International Conference on Artificial Intelligence Planning Systems, AIPS 1998*, 36–43. AAAI.

Di Tizio, G.; Speicher, P.; Simeonovski, M.; Backes, M.; Stock, B.; and Künnemann, R. 2023. Pareto-optimal defenses for the web infrastructure: Theory and practice. *ACM Transactions on Privacy and Security*, 26(2): 1–36.

Domshlak, C.; Hoffmann, J.; and Katz, M. 2015. Red-black planning: A new systematic approach to partial delete relaxation. *Artif. Intell.*, 221: 73–114.

Fikes, R.; and Nilsson, N. 1971. STRIPS: A New Approach to the Application of Theorem Proving to Problem Solving. *Artificial Intelligence*, 2(3–4): 189–208.

Hoffmann, J.; and Nebel, B. 2001. The FF Planning System: Fast Plan Generation Through Heuristic Search. *Journal of Artificial Intelligence Research (JAIR)*, 14: 2531–302.

Immerman, N. 1988. Nondeterministic space is closed under complementation. *SIAM Journal on computing*, 17(5): 935–938.

Jonsson, P.; and Bäckström, C. 1998. Tractable Plan Existence Does Not Imply Tractable Plan Generation. *Ann. Math. Artif. Intell.*, 22(3-4): 281–296.

Littman, M. L. 1997. Probabilistic Propositional Planning: Representations and Complexity. In *Proceedings of the Fourteenth National Conference on Artificial Intelligence AAAI 97*, 748–754. AAAI Press / The MIT Press.

Pham, F.; and Torralba, A. 2023. Can I Really Do That? Verification of Meta-Operators via Stackelberg Planning. In *Proceedings of the 32nd International Joint Conference on Artificial Intelligence (IJCAI 2023)*. IJCAI.

Rintanen, J. 1999. Constructing conditional plans by a theorem-prover. *Journal of Artificial Intelligence Research*, 10: 323–352.

Rintanen, J. 2004. Complexity of Planning with Partial Observability. In *Proceedings of the 14th International Conference on Automated Planning and Scheduling (ICAPS 2004)*, 345–354. AAAI Press.

Sauer, P.; Steinmetz, M.; Künnemann, R.; and Hoffmann, J. 2023. Lifted Stackelberg Planning. In *Proceedings of the 33rd International Conference on Automated Planning and Scheduling (ICAPS 2023)*. AAAI Press.

Speicher, P.; Steinmetz, M.; Backes, M.; Hoffmann, J.; and Künnemann, R. 2018a. Stackelberg planning: Towards effective leader-follower state space search. In *Proceedings of the 32nd AAAI Conference on Artificial Intelligence (AAAI 2018)*, 6286–6293. AAAI Press.

Speicher, P.; Steinmetz, M.; Künnemann, R.; Simeonovski, M.; Pellegrino, G.; Hoffmann, J.; and Backes, M. 2018b. Formally reasoning about the cost and efficacy of securing the email infrastructure. In *2018 IEEE European Symposium on Security and Privacy (EuroS&P)*, 77–91. IEEE.

Stockmeyer, L. J.; and Chandra, A. K. 1979. Provably Difficult Combinatorial Games. *SIAM J. Comput.*, 8(2): 151–174.

Szelepcsényi, R. 1987. The method of forcing for nondeterministic automata. *Bulletin European Association for Theoretical Computer Science*, 33: 96–100.

Torralba, Á.; Speicher, P.; Künnemann, R.; Steinmetz, M.; and Hoffmann, J. 2021. Faster Stackelberg Planning via Symbolic Search and Information Sharing. In *Proceedings of the 35th AAAI Conference on Artificial Intelligence (AAAI 2021)*, 11998–12006. AAAI Press.

Turner, H. 2002. Polynomial-length planning spans the polynomial hierarchy. In *Logics in Artificial Intelligence: 8th European Conference (JELIA 2002)*, 111–124. Springer.