Abstractions for Oversubscription Planning

Vitaly Mirkis
Advisor: **Carmel Domshlak**
Technion - Israel Institute of Technology
Haifa, Israel

Abstract

In deterministic oversubscription planning, the objective is to achieve an as valuable as possible subset of goals within a fixed allowance of the total action cost. Although numerous applications in various fields share this objective, no substantial algorithmic advances have been made beyond the very special settings of net-benefit optimization. Tracing the key sources of progress in classical planning, we identify a severe lack of domain-independent approximations for oversubscription planning, and start with investigating the prospects of abstraction approximations for this problem. In particular, we define the notion of additive abstractions for oversubscription planning, study the complexity of deriving effective abstractions from a rich space of hypotheses, and reveal some substantial, empirically relevant islands of tractability.

Introduction

In the context of deterministic planning, the basic structure of acting in situations with underconstrained or overconstrained resources is respectively captured by *classical* planning and *oversubscription* planning. In classical planning, all goals must be achieved at as low a total cost of the actions as possible. In oversubscription planning, an as valuable as possible subset of goals should be achieved within a fixed allowance of the total action cost. While both theory and practice of classical planning have been rapidly advancing, progress in oversubscription planning has been achieved mostly in the special setting of net-benefit planning, where no explicit restriction is put on the plan cost, the action costs and goal utilities are assumed to be comparable, and the objective is thus to maximize the difference between the cumulative value of the achieved goals and the cost invested in achieving them. Research on net-benefit planning resulted in numerous interesting algorithms, but recently it was shown that net-benefit planning is polynomial-time reducible to classical planning (Keyder and Geffner 2009), and thus constitutes an extremely special fragment of oversubscription planning.

A closer look at the recent progress in classical planning reveals that, to a large extent, it stems from advances in domain-independent approximations, or heuristics, of the cost needed to achieve all the goals from a given state. It is thus possible that having a similarly rich pallet of effective heuristic functions for oversubscription planning would advance the state-of-the-art in that setting of automated planning. In principal, the reduction of Keyder and Geffner (2009) from net-benefit to classical planning can be used to reduce oversubscription planning to classical planning with numeric state variables (Fox and Long 2003; Helmert 2002). So far, however, progress in classical planning with numeric state variables has mostly been achieved along variants of *monotonic*, or *delete-free*, *relaxation* heuristics (Hoffmann 2003; Edelkamp 2003), and these heuristics do not preserve information on consumable resources: the "negative" action effects that decrease the values of numeric variables are ignored, possibly up to some special handling of so-called "cyclic resource transfer" (Coles et al. 2008b).

In this work we make first steps towards effective heuristics for oversubscription planning, and in particular, towards admissible *abstraction heuristics* for this problem. In classical planning, state-space abstractions are among the most prominent foundations for devising admissible heuristics (Edelkamp 2002; Haslum et al. 2007; Helmert, Haslum, and Hoffmann 2007; Katz and Domshlak 2010a). Departing from the most basic question of what state-space abstractions for oversubscription planning actually are (and what they are not), we show that the very notion of abstraction substantially differs in classical and in oversubscription planning. We define additive abstractions and abstraction heuristics for oversubscription planning, and investigate computational complexity of deriving effective abstraction heuristics in the scope of homomorphic abstraction skeletons, paired with cost, value, and budget partitions. Along with revealing some significant islands of tractability, we expose an interesting interplay between knapsack-style problems, convex optimization, and principles borrowed from explicit abstractions for classical planning. We believe that this interplay opens the road to much further research.

Formalism and Background

In line with the SAS$^+$ formalism for deterministic planning (Bäckström and Klein 1991; Bäckström and Nebel 1995), a ***planning task structure*** is given by a pair $\Pi = \langle V, A \rangle$, where V is a set of n finite-domain *state variables*, and A is a finite set of *actions*. Each complete assignment to V is called a *state*, and $S = dom(v_1) \times \cdots \times dom(v_n)$ is the *state space* of Π. Each action a is a pair $\langle \mathsf{pre}(a), \mathsf{eff}(a) \rangle$

of partial assignments to V called *preconditions* and *effects*, respectively. Denoting by $\mathcal{V}(p) \subseteq V$ the subset of variables instantiated by a partial assignment p, action a is applicable in a state s iff $s[v] = \mathsf{pre}(a)[v]$ for all $v \in \mathcal{V}(\mathsf{pre}(a))$. Applying a changes the value of each $v \in \mathcal{V}(\mathsf{eff}(a))$ to $\mathsf{eff}(a)[v]$. The resulting state is denoted by $s[\![a]\!]$; by $s[\![\langle a_1, \dots, a_k\rangle]\!]$ we denote the state obtained from sequential application of the (applicable in turn) actions $a_1, \dots, a_k$ starting at state s.

In classical planning, a planning task $\Pi = \langle V, A; s_0, G, c\rangle$ extends its structure with an initial state $s_0 \in S$, a goal specification G, typically modeled as a partial assignment to V, and a real-valued, nonnegative action cost function $c : A \rightarrow \mathbb{R}^{0+}$. An action sequence ρ is called an s-plan if it is applicable in s, and $G \subseteq s[\![\rho]\!]$. An s-plan is optimal if the sum of its action costs is minimal among all s-plans. The objective in classical planning is to find an s_0-plan of as low cost as possible, while optimal classical planning is devoted to searching for optimal s_0-plans only.

In contrast, in oversubscription planning, a ***planning task*** $\Pi = \langle V, A; s_0, c, u, b\rangle$ extends its structure with four components: an *initial state* $s_0 \in S$ and an *action cost function* $c : A \rightarrow \mathbb{R}^{0+}$ as above, plus a succinctly represented and efficiently computable *state value* function $u : S \rightarrow \mathbb{R}^{0+}$, and a *cost budget* $b \in \mathbb{R}^{0+}$. An action sequence ρ is called an *s-plan* if it is applicable in s, and $\sum_{a\in\rho} c(a) \leq b$. An s-plan ρ is *optimal* if the value of the state reached from s via ρ is maximal among all s-plans; in this context, $h^*(s) = \max\{u(s') \mid s' \in S, c(s, s') \leq b\}$ is the value achievable by an optimal s-plan in Π. The objective in oversubscription planning is to find an s_0-plan that brings the system within the cost allowance to as valuable a state as possible, and ***optimal oversubscription planning*** is devoted to searching for optimal s_0-plans only.

Each planning task Π induces a state-transition model, often called its *transition graph*. A ***transition graph structure*** (or ***tg-structure***, for short) is a triplet $\mathcal{T} = \langle S, L, Tr\rangle$ where S is the finite set of states, L is the finite set of labels, and $Tr \subseteq S \times L \times S$ is a set of labeled state transitions. The tg-structure $\mathcal{T}(\Pi)$ induced by a planning task $\Pi = \langle V, A; s_0, c, u, b\rangle$ is induced by the structure $\langle V, A\rangle$ of the latter: the states and labels of $\mathcal{T}(\Pi)$ are states $S = dom(V)$ and actions A of Π, respectively, and $(s, a, s[\![a]\!]) \in Tr$ iff action a is applicable in state s.

In oversubscription planning, each tg-structure $\mathcal{T} = \langle S, L, Tr\rangle$ implicitly defines a ***space of performance measures*** that can be associated with it. In the context of oversubscription planning, this space constitutes $C \times U \times B$ where C is the set of all nonnegative real-valued functions from labels L, U is the set of all nonnegative real-valued functions from states S, and $B = \mathbb{R}^{0+}$. A ***transition graph*** (or ***t-graph***, for short) $\Phi = \langle \mathcal{T}, c, u, b\rangle$ associates a tg-structure $\mathcal{T}$ with a specific performance measure $(c, u, b) \in C \times U \times B$, and the t-graph induced by a planning task $\Pi = \langle V, A; s_0, c, u, b\rangle$ is $\Phi(\Pi) = \langle \mathcal{T}(\Pi), c, u, b\rangle$.

A few auxiliary notions: For $k \in \mathbb{N}^+$, by $[k]$ we denote the set $\{1, 2, \dots, k\}$. An s-path (ss'-path) in a t-graph $\Phi = \langle \mathcal{T}, c, u, b\rangle$ is a path in $\mathcal{T}$ from state s (to state s') along the transitions of $\mathcal{T}$. For an ss'-path π, by $\hat{u}(\pi)$ we refer to the value $u(s')$ of its endpoint. An s-path π is an s-plan for Φ if $\sum_{(s,l,s')\in\pi} c(l) \leq b$; $\mathbb{P}(s)$ is the set of all s-plans for Φ, and $\pi \in \mathbb{P}(s)$ is an optimal s-plan for Φ if $\pi = \mathrm{argmax}_{\pi'\in\mathbb{P}(s)}\, \hat{u}(\pi')$.

For a planning task Π, searching in its t-graph $\Phi(\Pi)$ corresponds to planning as state-space search. Informed such search procedures employ heuristic functions $h : S \rightarrow \mathbb{R}^{0+} \cup \{\infty\}$ to estimate the attractiveness of the graph region reachable from a given state s. For instance, in classical planning, heuristic h estimates the distance from the given state to the nearest goal state, and it is lower-bounding, or *admissible*, if $h(s) \leq h^*(s)$ for all states s. A useful heuristic function must be both efficiently computable from the planning task, as well as relatively accurate in its estimates. Improving the accuracy of a heuristic function without substantially worsening the time complexity of computing it translates into faster search for plans.

Unlike in oversubscription planning, numerous conceptual and computational ideas in classical planning have been translated to interesting heuristic functions. These ideas include monotonic relaxation (Bonet and Geffner 2001; 2001; Hoffmann and Nebel 2001), critical trees (Haslum and Geffner 2000), logical landmarks for goal reachability (Richter, Helmert, and Westphal 2008; Karpas and Domshlak 2009; Helmert and Domshlak 2009), and abstractions (Edelkamp 2001; Helmert, Haslum, and Hoffmann 2007; Katz and Domshlak 2010a). Also, different heuristics can be combined into their point-wise maximizing and additive ensembles (Edelkamp 2001; Haslum, Bonet, and Geffner 2005; Coles et al. 2008a; Katz and Domshlak 2010b). Unfortunately, while some of these ideas have also been translated to classical planning with numeric state variables, the resulting heuristics do not fit well to the specifics of oversubscription planning.

Abstractions for Oversubscription Planning

Our focus here is on abstractions for oversubscription planning, from their very definition and properties, to the prospects of deriving abstraction heuristics.

The term “abstraction" is usually associated with simplifying the original system, factoring out details less crucial in the given context. Which details can be factored out and which had better be preserved depends on the context. In classical planning, the abstract t-graphs are required not to increase the distances between the (abstracted) states (Edelkamp 2001; Helmert, Haslum, and Hoffmann 2007). Such “distance conservation” is in particular guaranteed by homomorphic abstractions, obtained by systematically contracting sets of states into single abstract states. In turn, an additive abstraction in classical planning is a set of abstractions, inter-constrained to *jointly* not overestimate the state-to-state costs of the original task. In contrast, a set of abstractions in oversubscription planning is constrained to jointly *not underestimate* the *value* that can be obtained from a *concrete state* of the original task within a given cost budget. Hence, the notions of additive abstraction for classical and oversubscription planning very different, with the latter providing us, for good and for bad, with many more degrees of freedom.

Definition 1 (Additive Abstraction) *Let $\Phi = \langle \mathcal{T}, c, u, b \rangle$ be a t-graph with $\mathcal{T} = \langle S, L, Tr \rangle$, and s_0 be a state in S.*

- *An **abstraction skeleton** $\mathcal{AS}$ of $\mathcal{T}$ is a set of pairs $\mathcal{AS} = \{\mathcal{T}_i, \alpha_i\}_{i \in [k]}$ where $\mathcal{T}_i = \langle S_i, L_i, Tr_i \rangle$ is a tg-structure and $\alpha_i : S \rightarrow S_i$ is a state mapping from $\mathcal{T}$ to $\mathcal{T}_i$.*
- *A set of t-graphs $\mathcal{A} = \{\Phi_i\}_{i \in [k]}$ is an **additive abstraction for** s_0 **in** Φ **with respect to** $\mathcal{AS}$, denoted by $\mathcal{A} \Subset_{s_0} \mathcal{AS}$, if*
  - *$\Phi_i = \langle \mathcal{T}_i, c_i, u_i, b_i \rangle$ and*
  - *if π is an optimal s_0-plan for Φ, and for $i \in [k]$, π_i is an optimal $\alpha_i(s_0)$-plan for Φ_i, then $\hat{u}(\pi) \leq \sum_{i \in [k]} \hat{u}_i(\pi_i)$.*

To illustrate this definition, consider a tg-structure $\mathcal{T} = \langle \{s_i\}_{i \in [5]}, \{l_i\}_{i \in [6]}, Tr \rangle$ depicted in Figure 1a, and an abstraction skeleton $\mathcal{AS} = \{(\mathcal{T}_1, \alpha_1), (\mathcal{T}_2, \alpha_2)\}$ of $\mathcal{T}$, with tg-structures $\mathcal{T}_1, \mathcal{T}_2$ as in Figure 1b and state mappings

$$\alpha_1(s_i) = \begin{cases} s_5^1, & i \in \{2, 4\} \\ s_i^1, & \text{otherwise} \end{cases} \quad \alpha_2(s_i) = \begin{cases} s_5^2, & i = 3 \\ s_i^2, & \text{otherwise.} \end{cases}$$

Let t-graphs $\Phi = \langle \mathcal{T}, c, u, b \rangle$, $\Phi_1 = \langle \mathcal{T}_1, c_1, u_1, b_1 \rangle$, $\Phi_2 = \langle \mathcal{T}_2, c_2, u_2, b_2 \rangle$ be defined via label cost functions c, c_1, c_2 that associate all labels with a cost of 1, budgets $b = b_1 = b_2 = 2$, and state value functions u, u_1, u_2 that evaluate to zero on all states except for the states s_5, s_5^1, s_5^2, on which they respectively evaluate to one. Considering the state s_1 in $\mathcal{T}$, the optimal s_1-plan for Φ is $\pi = \langle (s_1, l_2, s_3), (s_3, l_4, s_5) \rangle$ with $\hat{u}(\pi) = 1$. The optimal $\alpha_1(s_1^1)$-plan for Φ_1 is $\pi_1 = \langle (s_1^1, l_1^1, s_5^1) \rangle$ with $\hat{u}_1(\pi_1) = 1$, and the optimal $\alpha_2(s_1^2)$-plan for Φ_2 is $\pi_2 = \langle (s_1^2, l_2^2, s_5)^2 \rangle$, with $\hat{u}_2(\pi_2) = 1$. Since $\hat{u}(\pi) \leq \hat{u}_1(\pi_1) + \hat{u}_2(\pi_2)$, $\mathcal{A} = \{\Phi_1, \Phi_2\}$ is an additive abstraction for s_1 in Φ with respect to $\mathcal{AS}$.

Definition 2 (Abstraction Heuristics) *Let Π be a planning task with state space S, and $\mathcal{AS} = \{\mathcal{T}_i, \alpha_i\}_{i \in [k]}$ be an abstraction skeleton of $\mathcal{T}(\Pi)$.*

- *For each state $s \in S$, and each additive abstraction $\mathcal{A} = \{\Phi_i\}_{i \in [k]} \Subset_s \mathcal{AS}$, $h_{\mathcal{A}}(s) = \sum_{i \in [k]} \hat{u}_i(\pi_i)$ where π_i is an optimal $\alpha_i(s)$-plan for Φ_i.*
- *A function $h : S \rightarrow \mathbb{R}^{0+}$ is an $\mathcal{AS}$-**heuristic** for Π if, for each state $s \in S$, $h(s) = h_{\mathcal{A}}(s)$ for some additive abstraction $\mathcal{A} \Subset_s \mathcal{AS}$.*

Theorem 1 *For any planning task Π, any abstraction skeleton $\mathcal{AS}$ of $\mathcal{T}(\Pi)$, any state s of Π, and any $\mathcal{A} \Subset_s \mathcal{AS}$,*

(1) $\mathcal{A}$ provides an admissible estimate $h_{\mathcal{A}}(s) \geq h^*(s)$, and

(2) if the t-graphs of $\mathcal{A}$ are given explicitly, then $h_{\mathcal{A}}(s)$ can be computed in time polynomial in $||\Pi||$ and $||\mathcal{A}||$.

Sub-claim (1) in Theorem 1 is immediate from Definition 2, and it in particular implies that, for any $\mathcal{AS}$-heuristic h for Π, $h(s) \geq h^*(s)$. The proof of (2) in Theorem 1 is also straightforward : Let $\mathcal{A} = \{\Phi_i = \langle \mathcal{T}_i, c_i, u_i, b_i \rangle\}_{i \in [k]}$ be an additive abstraction for s in Π. For $i \in [k]$, let $S'_i = \{s' \in S_i \mid c_i(\alpha_i(s), s') \leq b_i\}$. Since $\mathcal{A}$ is given explicitly, computing shortest paths from $\alpha_i(s)$ to all states in $\mathcal{T}_i$, and thus computing S'_i, can be done in time polynomial in $||\mathcal{A}||$ for all $i \in [k]$. If π_i is an optimal plan for Φ_i, then by Definition 1, $\hat{u}_i(\pi_i) = \max_{s' \in S'_i} u_i(s')$, and thus computing $h_{\mathcal{A}}(s) = \sum_{i \in [k]} \hat{u}_i(\pi_i)$ is polynomial time in $||\mathcal{A}||$.

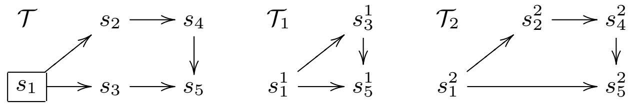

Figure 1: Illustration for our running example.

While Theorem 1 is positive, its tractability sub-claim establishes only a necessary condition for practical relevance of abstractions to oversubscription planning. First, note that the notion of abstraction in oversubscription planning is inherently state dependent: it is possible that, for a pair of states s, s' of a t-graph Φ and a set of t-graphs $\mathcal{A} = \{\Phi_i\}_{i \in [k]}$, it holds that $\mathcal{A} \Subset_s \mathcal{AS}$ while $\mathcal{A} \not\Subset_{s'} \mathcal{AS}$. Hence, given an abstraction skeleton $\mathcal{AS}$ for a task Π, and a state s of Π for which we need to estimate $h^*(s)$, first we have to *discover* an abstraction $\mathcal{A} \Subset_s \mathcal{AS}$. Second, different abstractions for s in Π with respect to the same abstraction skeleton $\mathcal{AS}$ will induce admissible estimates of very different quality. For instance, given a planning task $\Pi = \langle V, A; s_0, c, u, b \rangle$, consider a set of t-graphs $\mathcal{A} = \{\Phi_i = \langle \mathcal{T}_i, c_i, u_i, b_i \rangle\}_{i \in [k]}$ in which, for $i \in [k]$, $u_i(s') = \max_{s \in \Pi} u(s)$ for all $s' \in S_i$, and $c_i(l^i) = 0$ for all $l \in L_i$. This set of t-graphs is trivially an additive abstraction for *all* states s in Π, yet the estimate of $h^*(s)$ it provides is the loosest one possible.

Before we proceed with considering the complexity of deriving quality heuristics based on specific families of abstraction skeletons, we define some additional terminology. Let Π be a planning task, s be a state of Π, and $\mathcal{AS} = \{\mathcal{T}_i, \alpha_i\}_{i \in [k]}$ be an abstraction skeleton of $\mathcal{T}(\Pi)$. If $\mathbf{C} = \times C_i$, $\mathbf{U} = \times U_i$, and $\mathbf{B} = \times B_i$, then $\mathbf{A}(s) \subseteq \mathbf{C} \times \mathbf{U} \times \mathbf{B}$ *is a subspace of the joint performance measure space corresponding to the additive abstractions for s in Π with respect to $\mathcal{AS}$*, that is, $(\mathbf{c}, \mathbf{u}, \mathbf{b}) \in \mathbf{A}(s)$ iff $\mathcal{A}_{(\mathbf{c},\mathbf{u},\mathbf{b})} = \{\langle \mathcal{T}_i, \mathbf{c}[i], \mathbf{u}[i], \mathbf{b}[i] \rangle\}_{i \in [k]} \Subset_s \mathcal{AS}$. Note that $\mathbf{A}(s)$ is a subset of, but not a combinatorial rectangle in, $\mathbf{C} \times \mathbf{U} \times \mathbf{B}$. For instance, consider t-graph Φ, state s_1 of Φ, and abstraction skeleton $\mathcal{AS}$ from our running example. Let $\mathbf{c} \in \mathbf{C}$ be a cost function vector with both $\mathbf{c}[1]$ and $\mathbf{c}[2]$ being constant, unit-cost functions, and two performance measures $(\mathbf{c}, \mathbf{u}, \mathbf{b}), (\mathbf{c}, \mathbf{u}', \mathbf{b}') \in \mathbf{C} \times \mathbf{U} \times \mathbf{B}$ being defined via budget vectors $\mathbf{b} = \{\mathbf{b}[1] = 2, \mathbf{b}[2] = 0\}$ and $\mathbf{b}' = \{\mathbf{b}'[1] = 0, \mathbf{b}'[2] = 2\}$, and value function vectors $\mathbf{u}$ and $\mathbf{u}'$, with $\mathbf{u}[1], \mathbf{u}[2], \mathbf{u}'[1]$, and $\mathbf{u}'[2]$ evaluating to zero on all states except for $\mathbf{u}[1](s_5^1) = \mathbf{u}'[2](s_5^2) = 1$. It is easy to verify that $(\mathbf{c}, \mathbf{u}, \mathbf{b}), (\mathbf{c}, \mathbf{u}', \mathbf{b}') \in \mathbf{A}(s_1)$, yet $(\mathbf{c}, \mathbf{u}', \mathbf{b}), (\mathbf{c}, \mathbf{u}, \mathbf{b}') \notin \mathbf{A}(s_1)$.

Ultimately, given a planning task Π, an abstraction skeleton $\mathcal{AS} = \{\mathcal{T}_i, \alpha_i\}_{i \in [k]}$ of $\mathcal{T}(\Pi)$, and a state s of Π, we would like to compute the optimal estimate $h_{\mathcal{AS}}(s) = \min_{\mathcal{A} \Subset_s \mathcal{AS}} h_{\mathcal{A}}(s)$. However, if, for instance, we are *given* a vector of value functions $\mathbf{u}$ that is *known* to belong to the projection of $\mathbf{A}(s)$ on $\mathbf{U}$, then we can search for a quality abstraction from the abstraction subset $H_{(-,\mathbf{u},-)}(s) \subset \mathbf{A}(s)$, corresponding to the projection of $\mathbf{A}(s)$ on $\{\mathbf{u}\}$. As we

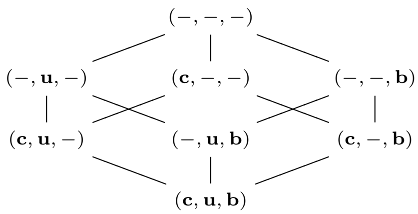

Figure 2: Fragments of restricted optimization over $\mathbf{A}(s)$.

show below, even some constrained estimate optimizations of this kind can be challenging. The lattice in Figure 2 depicts the whole range of options for such constrained optimization; at the extreme settings, $H_{(-,-,-)}(s)$ is simply a renaming of $\mathbf{A}(s)$, and $h_{(\mathbf{c},\mathbf{u},\mathbf{b})}(s)$ corresponds to a single abstraction $(\mathbf{c},\mathbf{u},\mathbf{b}) \in \mathbf{A}(s)$.

In a recent work (Domshlak and Mirkis 2013) we present some theoretical and practical initial results.

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. 2001. Planning as heuristic search. *Artificial Intelligence* 129(1–2):5–33.

Coles, A. I.; Fox, M.; Long, D.; and Smith, A. J. 2008a. Additive-disjunctive heuristics for optimal planning. In *Proceedings of the 18th International Conference on Automated Planning and Scheduling (ICAPS)*, 44–51.

Coles, A. I.; Fox, M.; Long, D.; and Smith, A. J. 2008b. A hybrid relaxed planning graph-LP heuristic for numeric planning domains. In *Proceedings of the 18th International Conference on Automated Planning and Scheduling (ICAPS)*.

Domshlak, C., and Mirkis, V. 2013. Abstractions for oversubscription planning. In *Proceedings of the 23rd International Conference on Automated Planning and Scheduling (ICAPS)*.

Edelkamp, S. 2001. Planning with pattern databases. In *Proceedings of the European Conference on Planning (ECP)*, 13–34.

Edelkamp, S. 2002. Symbolic pattern databases in heuristic search planning. In *Proceedings of the International Conference on AI Planning and Scheduling (AIPS)*, 274–293.

Edelkamp, S. 2003. Taming numbers and durations in the model checking integrated planning system. *Journal of Artificial Intelligence Research* 20:195–238.

Fox, M., and Long, D. 2003. PDDL2.1: An extension to PDDL for expressing temporal planning problems. *Journal of Artificial Intelligence Research* 20:61–124.

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.

Haslum, P.; Botea, A.; Helmert, M.; Bonet, B.; and Koenig, S. 2007. Domain-independent construction of pattern database heuristics for cost-optimal planning. In *Proceedings of the 19th National Conference on Artificial Intelligence (AAAI)*, 1007–1012.

Haslum, P.; Bonet, B.; and Geffner, H. 2005. New admissible heuristics for domain-independent planning. In *Proceedings of the 20th National Conference on Artificial Intelligence (AAAI)*, 1163–1168.

Helmert, M., and Domshlak, C. 2009. Landmarks, critical paths and abstractions: Whatï£¡s the difference anyway? In *under review for the 19th International Conference on Automated Planning and Scheduling (ICAPS)*.

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. 2002. Decidability and undecidability results for planning with numerical state variables. In *Proceedings of the Sixth International Conference on Artificial Intelligence Planning and Scheduling*.

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

Hoffmann, J. 2003. The Metric-FF planning system: Translating "ignoring delete lists" to numeric state variables. *Journal of Artificial Intelligence Research* 20:291–341.

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. 2010a. Implicit abstraction heuristics. *Journal of Artificial Intelligence Research* 39:51–126.

Katz, M., and Domshlak, C. 2010b. Optimal admissible composition of abstraction heuristics. *Artificial Intelligence* 174:767–798.

Keyder, E., and Geffner, H. 2009. Soft goals can be compiled away. *Journal of Artificial Intelligence Research* 36:547–556.

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