$h^m(P) = h^1(P^m)$: Alternative Characterisations of the Generalisation From $h^{\max}$ To h^m

Patrik Haslum
Australian National University
patrik.haslum@anu.edu.au

Abstract

The h^m ($m = 1, \ldots$) family of admissible heuristics for STRIPS planning with additive costs generalise the $h^{\max}$ heuristic, which results when $m = 1$. We show that the step from h^1 to h^m can be made by changing the planning problem instead of the heuristic function. This furthers our understanding of the h^m heuristic, and may inspire application of the same generalisation to admissible heuristics stronger than $h^{\max}$. As an example, we show how it applies to the additive variant of h^m obtained via cost splitting.

Introduction

The $h^{\max}$ heuristic, introduced by Bonet and Geffner in 1999, is the first of the "new generation" of admissible heuristics for (propositional STRIPS) planning with additive action costs, and it is certainly the most widely known and understood. The relaxation underlying the $h^{\max}$ heuristic is taking the cost of achieving a conjunction of atoms to equal the cost of achieving the most costly single atom in the conjunction. This assumption makes the heuristic simple, conceptually and computationally, often allowing more advanced admissible heuristics to be related to $h^{\max}$ (e.g. Helmert and Domshlak 2009). However, it also limits the power of the heuristic. In particular, the $h^{\max}$ heuristic is invariant under delete relaxation and therefore bounded above by h^+, the optimal delete-relaxation heuristic. This implies, for instance, that the $h^{\max}$ estimate can not include the cost of any action more than once, even though an action may be needed an exponential number of times in a plan.

The h^m ($m = 1, \ldots$) family of admissible heuristics is based on the same relaxation as $h^{\max}$, but parameterise it by the maximum size m of conjunctions considered. Thus $h^{\max} = h^1$. This makes the h^m heuristics more powerful. For instance, for $m > 1$, h^m is not bounded by h^+, and even equals the real optimal cost function h^* for sufficiently large m. However, the complexity of computing the h^m heuristic rises exponentially with m, and for practical purposes it is typically limited to $m = 2$ or $m = 3$. Thus, a case can be made for seeking ways to generalise admissible heuristics more powerful than $h^{\max}$ in a maner analogous to the way h^m generalises $h^{\max}$.

This paper presents a new characterisation of the h^m heuristics, by showing that h^m can be obtained as h^1 (i.e., $h^{\max}$) over a modified planning problem. That is, for a planning problem P we construct a new problem P^m and show that $h^m(P) = h^1(P^m)$. The size of P^m is polynomial in the size of P (though exponential in m). However, P^m does not preserve real plan costs, i.e., $h^*(P^m)$ may be greater than $h^*(P)$. This has the undesirable implication that applying an admissible heuristic to P^m does not necessarily yield an estimate that is admissible for P. Alternative constructions are also sketched.

The new characterisation does not directly lead to a practical way of generalising an arbitrary admissible heuristic from 1 to m. Nor is it a more efficient way to compute h^m: computing $h^1(P^m)$ typically requires more time and memory than computing $h^m(P)$.

What it does offer is some new insight into the mechanics of the h^m heuristic, which, hopefully, suggests how the step from 1 to m can be carried out. As an example, we apply this insight to generalise Helmert's and Domshlak's (2009) cost partitioning scheme for additive h^1 to additive h^m.

Background

We adopt the standard definition of a propositional STRIPS planning problem, without negation in action preconditions or the goal (see, e.g., Ghallab, Nau, and Traverso 2004, chapter 2). By a (*conjunctive*) *condition*, c, we mean a conjunction of atoms in the vocabulary of planning problem P, without repetition. We treat conditions as sets of atoms and use standard set notations for them. The *size* of a condition, $|c|$, is the number of atoms in it. A sequence of actions (or *plan*) *achieves* condition c (from state s) iff the sequence is executable in s and leads to a state where c holds. We assume an additive cost objective, i.e., each action a has a non-negative cost, $\mathrm{cost}(a)$, and the cost of a plan is the sum of the cost of actions in it. The optimal cost function, $h^*(c)$, is defined as the minimum cost over all plans for c from the initial state of P, with $h^*(c) = \infty$ if no plan for c exists.

The h^m Heuristic

The h^m heuristic ($m \geq 1$) is defined as the greatest (w.r.t point-wise comparison) fixed point solution to the recursive equation

$$h^m(P,c) = \begin{cases} 0 & \text{if } c \text{ holds in } s_I \\ \min_a h^m(P,\mathrm{R}(c,a)) + \mathrm{cost}(a) & \text{if } |c| \leqslant m \\ \max_{c' \subseteq c, |c'| \leqslant m} h^m(P,c') & \text{if } |c| > m \end{cases} \quad (1)$$

where s_I is the initial state of P, $\mathrm{R}(c,a)$ is the *regression operator*, defined as

$$\mathrm{R}(c,a) = (c - \mathrm{add}(a)) \cup \mathrm{pre}(a)$$

if $\mathrm{del}(a) \cap c = \emptyset$ and undefined otherwise, and the minimum in the second case of equation (1) is over actions a such that $\mathrm{R}(c,a)$ is defined; if there is no such action, the minimum is taken to be ∞. The reference to the problem, P, is omitted when it is clear from context. Conversely, to refer to the h^m function as such, while distinguishing the problem over which it is defined, we drop the second argument.

The h^m heuristic is admissible, meaning that $h^m(c) \leq h^*(c)$ for every c. Moreover, it is (non-strictly) increasing in m, i.e., $h^m(c) \leq h^{m+i}(c)$ when $i \geq 0$, and for every planning problem P there exists an m', bounded by the number of atoms in P, such that $h^{m'}(P) = h^*(P)$ (Haslum 2006).

A Procedural Characterisation of h^m

In the presence of actions with zero cost, equation (1) does not necessarily have a unique fix point solution. By defining the h^m function as the point-wise greatest fix point of equation (1), values that are not uniquely determined are effectively assigned ∞. For proving the equivalence that is the main topic of this paper, the following, more procedural, characterisation of the h^m function is useful.

Definition 1 *The series of functions* $\{h^m_i\}_{i=0,\ldots}$ *is defined by* $h^m_0(c) =$

$$\begin{cases} 0 & \textit{if } c \textit{ holds in } s_I \\ \infty & \textit{otherwise} \end{cases} \quad (2)$$

and $h^m_{i+1}(c) =$

$$\begin{cases} \min(h^m_i(c), \\ \quad \min_a h^m_i(\mathrm{R}(c,a)) + \mathrm{cost}(a)) & \textit{if } |c| \leqslant m \\ \max_{c' \subseteq c, |c'| \leqslant m} h^m_{i+1}(c') & \textit{if } |c| > m \end{cases} \quad (3)$$

where, as before, the minimum is over actions a *such that* $\mathrm{R}(c,a)$ *is defined, and* ∞ *if there is no such action. The series converges at* i *iff* $h^m_i = h^m_{i-1}$.

This definition captures the generalised Bellman-Ford (GBF) procedure for computing h^m. The GBF procedure is a label correcting algorithm: it assigns a crude initial cost estimate (0 or ∞) to every condition and then iteratively applies local updates until costs converge (Haslum 2006). Function h^m_i is the result after i iterations.

Theorem 2 *The series* $\{h^m_i\}_{i=0,\ldots}$ *converges to* h^m.
Proof (sketch): (*i*) The series converges.
If the series does not converge at i there must be some c, with $|c| \leq m$, such that $h^m_{i+1}(c) \neq h^m_i(c)$. The value of $h^m_{i+1}(c)$ can only be less than $h^m_i(c)$, due to minimisation with h^m_i. The new value is a sum of action costs, and thus can not be negative, nor less than $h^m_i(c)$ by an arbitrarily small amount. Hence, an infinite number of iterations is impossible.
(*ii*) If the series converges at i, h^m_i satisfies equation (1).
If c holds in s_I, the initial state of P, $h^m_0(c) = 0$, and thus $h^m_i(c) = 0$ since values only decrease. Let c be a condition with $|c| \leq m$ that does not hold in s_I. Since the series has converged at i, $\min_a h^m_i(\mathrm{R}(c,a)) + \mathrm{cost}(a) = \min_a h^m_{i-1}(\mathrm{R}(c,a)) + \mathrm{cost}(a) \geq h^m_i(c)$. $h^m_i(c) < \min_a h^m_i(\mathrm{R}(c,a)) + \mathrm{cost}(a)$ implies $h^m_{i-1}(c) < \min_a h^m_{i-1}(\mathrm{R}(c,a)) + \mathrm{cost}(a)$, and thus, by induction, $h^m_0(c) < \min_a h^m_0(\mathrm{R}(c,a)) + \mathrm{cost}(a)$, which can be only if $h^m_0(c) = 0$, which it is only if c holds in s_I, contrary to assumption.
(*iii*) No function greater than h^m_i at any point can satisfy equation (1).
$h^m_0(c)$ has the greatest value that may possibly satisfy equation (1): it is ∞ unless it must be 0. Whenever $h^m_{i+1}(c)$ is less than $h^m_i(c)$ it is upperbounded by $\min_a h^m_i(\mathrm{R}(c,a)) + \mathrm{cost}(a)$, as required by equation (1).

Items (*i*)–(*iii*) combined imply that the function h^m_i at convergence is the (point-wise) greatest fix point solution to equation (1) and therefore equal to h^m. □

The Additive h^m Heuristic

For computationally feasible values of m, the h^m heuristic is often too weak. For planning with additive costs, a stronger additive heuristic can be obtained via the "cost splitting" method. This is a simple and general method, applicable to any admissible heuristic for problems with additive costs, that has been used by many researchers (e.g., Edelkamp 2001; Haslum, Bonet, and Geffner 2005; Katz and Domshlak 2008,Yang et al. 2008). The formulation below is essentially that of Katz and Domshlak (2008).

Let P be a planning problem: A *cost function for* P is a function that maps actions in P to the domain of action costs.[1] If C is a cost function for P, $C(P)$ (C *applied to* P) denotes a planning problem exactly like P except that the cost of each action a in $C(P)$ is $C(a)$. A collection of cost functions, $C_1, \ldots, C_k$, is an *admissible cost partitioning for* P iff $\left(\sum_{i=1,\ldots,k} C_i(a)\right) \leq \mathrm{cost}(a)$ for each action a.

Theorem 3 *Let* P *be a planning problem,* $C_1, \ldots, C_k$ *an admissible cost partitioning for* P*, and* $h_1, \ldots, h_k$ *a collection of functions such that* h_i *is an admissible heuristic for* $C_i(P)$*. Then* $h_\Sigma = \sum_{i=1,\ldots,k} h_i$ *is an admissible heuristic for* P.

The additive h^m heuristic is obtained by instantiating each of the h_i's in theorem 3 with $h^m(C_i(P))$.

The P^m Construction

Given a planning problem P and $m > 1$, we construct a problem P^m such that $h^m(P) = h^1(P^m)$. The size of P^m is polynomial in the size of P. However, $h^*(P^m)$ can be

[1] The intrinsic action cost assignment, $\mathrm{cost}(a)$, may be viewed as but another cost function for P.

greater than $h^*(P)$. We show that the additive h^1 heuristic applied to P^m, under an additional constraint on the splitting of action costs, remains admissible for P.

Atoms in P^m correspond to conjunctive conditions of size at most m over the atoms of P. Likewise, actions in P^m correspond to sets, containing one "regular" action from P and a number of "no-op" actions. To reduce confusion, we will refer to the atoms and actions in problem P^m as *meta-atoms* and *meta-actions*, respectively. Note, however, that P^m is a perfectly ordinary propositional STRIPS planning problem: In particular, theorem 2 applies to it.

Definition 4 *Let P be a propositional STRIPS problem (without negation). The problem P^m contains a meta-atom π_c for each conjunctive condition c of size at most m over the atoms of P.*

For each action a in P, and for each set f of at most $m-1$ atoms such that f is disjoint from $\mathrm{add}(a)$ and from $\mathrm{del}(a)$, P^m contains a meta-action $\alpha_{a,f}$ with:

$$\begin{aligned}
&\mathrm{pre}(\alpha_{a,f}) = \{\pi_c \mid c \subseteq (\mathrm{pre}(a) \cup f), |c| \leq m\} \\
&\mathrm{add}(\alpha_{a,f}) = \\
&\quad \{\pi_c \mid c \subseteq (\mathrm{add}(a) \cup f), (c \cap \mathrm{add}(a)) \neq \emptyset, |c| \leq m\} \\
&\mathrm{del}(\alpha_{a,f}) = \emptyset
\end{aligned}$$

and $\mathrm{cost}(\alpha_{a,f}) = \mathrm{cost}(a)$.

The initial state of P^m, σ_I, assigns true to every meta-atom π_c ($|c| \leq m$) such that c holds in the initial state s_I of P, and false to every other meta-atom. The goal condition of P^m is $\Gamma = \{\pi_c \mid c \subseteq G, |c| \leq m\}$, where G is the goal condition of P.

There is a straightforward correspondence between (conjunctive) conditions over atoms in P and conditions over meta-atoms in P^m: if $c = \{p_{i_1}, \ldots, p_{i_k}\}$ is a condition over the vocabulary of P, the corresponding condition in P^m is $\gamma(c) = \{\pi_{c'} \mid c' \subseteq c, |c'| \leq m\}$; conversely, if $\gamma = \{\pi_{c_1}, \ldots, \pi_{c_k}\}$ is a conjunction over the vocabulary of P^m, the corresponding condition in P is $c(\gamma) = \bigcup_{\pi_{c_i} \in \gamma} c_i$. The goal of P^m corresponds to the goal of P.

As expected, $c(\gamma(c)) = c$. However, $\gamma(c(\gamma)) \supseteq \gamma$. This is because $\gamma(c)$ contains *all* meta-atoms corresponding to conditions of size at most m over the set of atoms in c, which need not be the case for an arbitrary condition over the vocabulary of P^m. There is an analogous correspondence between states of P and certain states of P^m.

Each action $\alpha_{a,f}$ in P^m corresponds to, in P, executing action a while simultaneously *preserving* the truth of each atom in the set f. (Note that regressing condition c through action a can be seen as planning to achieve atoms in $c \cap \mathrm{add}(a)$ by a, while preserving the remaining, already achieved atoms in $c - \mathrm{add}(a)$.) This "composite" action requires every atom in $\mathrm{pre}(a) \cup f$ to hold before execution, and hence every meta-atom corresponding to a condition that is a subset (of size at most m) of this set is in $\mathrm{pre}(\alpha_{a,f})$. The positive effects of $\alpha_{a,f}$ include each meta-atom corresponding to a conjunction of one or more atoms made true by a, with the remaining conjuncts atoms in f. All such conjunctions will hold in the state resulting from executing a while preserving f. Because our only interest in the problem P^m is applying to it the h^1 heuristic, which is invariant under delete relaxation, actions in P^m have empty delete sets. Note that although each action a in problem P is represented by (potentially) several meta-actions in P^m, each of those meta-actions carries the full cost of a.

Theorem 5 *Let P be a planning problem and c a conjunctive condition over the vocabulary of P: $h^m(P,c) = h^1(P^m, \gamma(c))$.*

Proof: We show by induction that $h^m_i(P,c) = h^1_i(P^m, \gamma(c))$, for $i \geq 0$. By theorem 2 above, the series converge to $h^m(P,c)$ and $h^1(P^m, \gamma(c))$, respectively, which are therefore equal.

It is easy to see that $\gamma(c) = \{\pi_{c'} \mid c' \subset c, |c| \leq m\}$ holds in the initial state σ_I of P^m iff c holds in the initial state s_I of P. Thus $h^m_0(P,c) = h^1_0(P^m, \gamma(c))$. Assuming the equality holds for $j < i$, consider a condition c with $|c| \leq m$. Then

$$\begin{aligned}
&h^m_i(P,c) = \\
&\quad \min\left(h^m_{i-1}(P,c), \min_a h^m_{i-1}(P, \mathrm{R}(c,a)) + \mathrm{cost}(a)\right).
\end{aligned} \tag{4}$$

Because $|c| \leq m$, $\gamma(c) = \{\pi_c\}$, i.e., there is a single meta-atom corresponding to c. Thus,

$$\begin{aligned}
&h^1_i(P^m, \gamma(c)) = \\
&\quad \min(h^1_{i-1}(P^m, \gamma(c)), \\
&\qquad \min_\alpha h^1_{i-1}(P^m, \mathrm{R}(\gamma(c), \alpha)) + \mathrm{cost}(\alpha).
\end{aligned} \tag{5}$$

The first terms in the outer minimums in (4) and (5), $h^m_{i-1}(P,c)$ and $h^1_{i-1}(P^m, \gamma(c))$, are equal by inductive assumption.

Let a be an action that attains the (inner) minimum in (4) and let $f = c - \mathrm{add}(a)$: the meta-action $\alpha_{a,f}$ adds π_c (and does not delete it), and, moreover, $\mathrm{R}(c,a) = (c - \mathrm{add}(a)) \cup \mathrm{pre}(a)$, and $\mathrm{R}(\gamma(c), \alpha_{a,f}) = (\{\pi_c\} - \mathrm{add}(\alpha_{a,f})) \cup \mathrm{pre}(\alpha_{a,f}) = \mathrm{pre}(\alpha_{a,f})$. By construction of $\alpha_{a,f}$, and the choice of f, $\gamma(\mathrm{R}(c,a)) = \mathrm{R}(\gamma(c), \alpha_{a,f})$. Since $h^1_{i-1}(P^m, \mathrm{R}(\gamma(c), \alpha_{a,f})) = h^m_{i-1}(P, \mathrm{R}(c,a))$ by inductive assumption, and $\mathrm{cost}(\alpha_{a,f}) = \mathrm{cost}(a)$ by construction, $h^1_i(P^m, \gamma(c))$ is no greater than $h^m_i(P,c)$. Conversely, let α be a meta-action that attains the (inner) minimum in (5): α is composed of an action a in P and a set f of preserved atoms, such that $\mathrm{del}(a) \cap c = \emptyset$ (since $\mathrm{R}(\{\pi_c\}, \alpha)$ is defined), $\mathrm{add}(a) \cap c \neq \emptyset$, and $c - \mathrm{add}(a) \subseteq f$. Furthermore, $c(\mathrm{R}(\{\pi_c\}, \alpha)) = \mathrm{pre}(a) \cup f \supseteq pre(a) \cup (c - \mathrm{add}(a)) = \mathrm{R}(c,a)$. Hence, by inductive assumption, and the property that $h^m_i(c') \geq h^m_i(c)$ whenever $c' \supseteq c$, $h^m_i(P,c)$ is no greater than $h^1_i(P^m, \gamma(c))$.

For any condition c with $|c| > m$, $h^m_i(P,c) = \max_{c' \subset c, |c| \leq m} h^m(P, c')$. Also, $\gamma(c) = \{\pi_{c'} \mid c' \subset c, |c| \leq m\}$, so $|\gamma(c)| > 1$; thus $h^1_i(P^m, \gamma(c)) = \max_{\pi_{c'} \in \gamma(c)} h^1_i(P^m, \{\pi_{c'}\})$. As the two maximums range over the same set of alternatives (conditions of size at most m over P), and equality of the two functions for this case has been shown above, they too are equal. □

The reason why $h^*(P^m)$ does not equal $h^*(P)$ is that each meta-action $\alpha_{a,f}$ adds *only* meta-atoms corresponding to conditions that combine atoms made true by the action a

and preserved atoms belonging to f, which is limited to at most $m-1$ atoms, while in real execution of an action any number of atoms may remain true by inertia. The following example illustrates.

Example 6 Consider a small Logistics problem, P_{Ex6}, with two packages (`P1` and `P2`) loaded in a truck (`T`) at location `A`. Executing `(drive T A B)` leads to a state where condition $c = \{$`(in P1 T)`, `(in P2 T)`, `(at T B)`$\}$ holds.

The corresponding condition in problem ${P_{\text{Ex6}}}^2$, $\gamma(c)$, contains a meta-atom $\pi_{\{p,q\}}$ for every pair of atoms $p, q \in c$ and a meta-atom $\pi_{\{p\}}$ for every single atom $p \in c$. However, there is no meta-action in ${P_{\text{Ex6}}}^2$ corresponding to executing the `(drive T A B)` action while preserving the truth of both atoms `(in P1 T)` and `(in P2 T)`, because meta-actions in ${P_{\text{Ex6}}}^2$ are constructed with f-sets of size 1. For example, $\alpha_{\texttt{(drive T A B)},\{\texttt{(in P1 T)}\}}$ adds $\pi_{\{\texttt{(at T B)}\}}$ and $\pi_{\{\texttt{(at T B)},\texttt{(in P1 T)}\}}$, but not $\pi_{\{\texttt{(at T B)},\texttt{(in P2 T)}\}}$. Thus, to achieve $\gamma(c)$, both $\alpha_{\texttt{(drive T A B)},\{\texttt{(in P1 T)}\}}$ and $\alpha_{\texttt{(drive T A B)},\{\texttt{(in P2 T)}\}}$ must be executed.

As a consequence of this fact, applying an arbitrary admissible heuristic h to P^m does not necessarily yield an estimate that is admissible for P.

Application To Additive h^m

Although the admissibility condition for cost partitionings is the same no matter which admissible heuristic it is applied to, what makes a *good* cost partitioning depends very much on the heuristic.

Let P be a planning problem and $C_1, \ldots, C_k$ a collection of cost functions for P^m satisfying

$$\sum_{i=1,\ldots,k} \left(\max_{\alpha_{a,f}} C_i(\alpha_{a,f}) \right) \leq \text{cost}(a) \tag{6}$$

for each action a in P (note that the maximum is over meta-actions representing a). Define the cost function $C_i^{\max}$ for P as $C_i^{\max}(a) = \max_{\alpha_{a,f}} C_i(\alpha_{a,f})$. The constraint on the C_i's (eq. 6) ensures that the collection $C_1^{\max}, \ldots, C_k^{\max}$ is an admissible cost partitioning for P. Thus, by theorem 3, $h_\Sigma^m = \sum_{i=1,\ldots,k} h^m(C_i^{\max}(P))$ is admissible for P.

Helmert and Domshlak (2009) describe a procedure that automatically creates a good admissible cost partitioning for the additive h^1 heuristic. The resulting additive heuristic is guaranteed to be at least as good as h^1, though this only for the goal condition.

By enforcing the extra constraint (6), the same procedure applied to P^m can be used to obtain cost partitionings for h^m. The required modification is straightforward, though the guarantee given by the original procedure is lost. However, accuracy of the resulting heuristic h_Σ^m can still be lowerbounded:

Theorem 7 $h_\Sigma^1 = \sum_{i=1,\ldots,k} h^1(C_i(P^m)) \leq h_\Sigma^m$.

Proof: Extend each $C_i^{\max}$ to a cost function for P^m by $C_i^{\max}(\alpha_{a,f}) = C_i^{\max}(a)$. Clearly, $h^1(C_i^{\max}(P^m)) \geq h^1(C_i(P^m))$, so $\sum_{i=1,\ldots,k} h^1(C_i^{\max}(P^m)) \geq h_\Sigma^1$. By construction, $C_i^{\max}(P^m) = C_i^{\max}(P)^m$ (recall that each representative $\alpha_{a,f}$ of action a in P has the same cost in P^m as a does in P). By theorem 5, $h^1(C_i^{\max}(P)^m) = h^m(C_i^{\max}(P))$, so $\sum_{i=1,\ldots,k} h^1(C_i^{\max}(P^m)) = h_\Sigma^m$. □

A corollary to the above is that h_Σ^1 is also admissible for P.

Other Constructions

From the preceding discussion (example 6 in particular) it should be clear that a problem P_*^m satisfying both $h^m(P) = h^1(P_*^m)$ and $h^*(P) = h^*(P_*^m)$ can be obtained simply by not limiting the size of f-sets in the construction of meta-actions (and defining their delete sets appropriately). Unfortunately, the size of P_*^m is exponential in the size of P as well as in m. The same problem can be written compactly if use of conditional effects is permitted: each meta-action α_a then combines the effects of action a with a conditional effect emulating a no-op for each atom $p \notin (\text{del}(a) \cup \text{add}(a))$. P_*^m may be seen as the result of compiling away those conditional effects, using the exponential-size but plan length-preserving compilation scheme (Nebel 2000). Alternatively, using the polynomial-size compilation scheme with linear increase in plan length results in a construction that preserves real plan costs but not exact h^m values. However, it does preserve information about unreachability (infinite h^m values).

References

Bonet, B., and Geffner, H. 1999. Planning as heuristic search: New results. In *Proc. 5th European Conference on Planning (ECP'99)*, 360–372.

Edelkamp, S. 2001. Planning with pattern databases. In *Proc. 6th European Conference on Planning (ECP'01)*, 13–24.

Ghallab, M.; Nau, D.; and Traverso, P. 2004. *Automated Planning: Theory and Practice*. Morgan Kaufmann Publishers. ISBN: 1-55860-856-7.

Haslum, P.; Bonet, B.; and Geffner, H. 2005. New admissible heuristics for domain-independent planning. In *Proc. 20th National Conference on AI (AAAI'05)*, 1163–1168.

Haslum, P. 2006. *Admissible Heuristics for Automated Planning*. Ph.D. Dissertation, Linköpings Universitet.

Helmert, M., and Domshlak, C. 2009. Landmarks, critical paths and abstractions: What's the difference anyway? In *Proc. 19th International Conference on Automated Planning and Scheduling (ICAPS'09)*.

Katz, M., and Domshlak, C. 2008. Optimal additive composition of abstraction-based admissible heuristics. In *Proc. of the 18th International Conference on Automated Planning and Scheduling (ICAPS'08)*.

Nebel, B. 2000. On the compilability and expressive power of propositional planning formalisms. *Journal of AI Research* 12:271–315.

Yang, F.; Culberson, J.; Holte, R.; Zahavi, U.; and Felner, A. 2008. A general theory of additive state space abstractions. *Journal of AI Research* 32:631–662.