LM-Cut Heuristic

Daniel Fišer

March 13, 2017

STRIPS

A problem in STRIPS is a tuple $\Pi = \langle \mathcal{F}, \mathcal{O}, s_{init}, s_{goal}, c \rangle$

- $\mathcal{F}$ is a finite set of **facts**
- $s_{init} \subseteq \mathcal{F}$ is an **initial state**; $s_{goal} \subseteq \mathcal{F}$ is a **goal specification**
- $\mathcal{O}$ is a set of well-formed **operators** o specified by $\mathrm{pre}(o) \subseteq \mathcal{F}$, $\mathrm{del}(o) \subseteq \mathcal{F}$, and $\mathrm{add}(o) \subseteq \mathcal{F}$
- $c : \mathcal{O} \mapsto \mathbb{R}_0^+$ is a cost function.

- A **state** $s \subseteq \mathcal{F}$; a **goal state** $s_g \supseteq s_{goal}$.
- An operator o is **applicable** in a state s iff $\mathrm{pre}(o) \subseteq s$ and the **resulting state** of applying o is $o[s] = (s \setminus \mathrm{del}(o)) \cup \mathrm{add}(o)$.
- A **sequence of operators** $\pi = \langle o_1, ..., o_n \rangle$: $\pi[s] = o_n[...o_2[o_1[s]]...]$.
- π is a **plan** iff $s_{goal} \subseteq \pi[s_{init}]$.

A **relaxed problem** Π^+ is derived from Π by ignoring all delete effects.

(Action/Operator) Landmark

Definition

A landmark $L \subseteq \mathcal{O}$ is a set of operators such that every plan π must contain at least one operator from L.

This type of landmark is sometimes also called action landmark or disjunctive action landmark.

$\mathrm{h}^{\max}_{c_i}$

$\mathrm{h}^{\max}_{c_i}$(s) for a state s and a cost function c_i

$$\forall f \in s : \mathrm{h}^{\max}_{c_i}(f) = 0$$

$$\forall f \in \mathcal{F} \setminus s : \mathrm{h}^{\max}_{c_i}(f) = \min_{o \in \mathcal{O}, f \in \mathrm{add}(o)} (\mathrm{h}^{\max}_{c_i}(o) + c_i(o))$$

$$\forall o \in \mathcal{O} : \mathrm{h}^{\max}_{c_i}(o) = \max_{f \in \mathrm{pre}(o)} \mathrm{h}^{\max}_{c_i}(f)$$

$$\mathrm{h}^{\max}_{c_i}(s) = \max_{f \in s_{goal}} \mathrm{h}^{\max}_{c_i}(f)$$

$\mathrm{h}^{\max}_{c_i}$

Algorithm 1: $\mathrm{h}^{\max}_{c_i}$

Input: $\Pi = \langle \mathcal{F}, \mathcal{O}, s_{init}, s_{goal}, c_i \rangle$, state s
Output: $\mathrm{h}^{\max}_{c_i}(f) \;\; \forall f \in \mathcal{F}$

Initialize min priority queue PQ.init($\{(f, 0) \mid f \in s\} \cup \{(f, \infty) \mid f \in \mathcal{F} \setminus s\}$) ;
Initialize number of unsatisfied preconditions: $U(o) \leftarrow |\mathrm{pre}(o)| \;\; \forall o \in \mathcal{O}$;
while *not PQ.empty()* **do**
    $(f, \mathrm{h}^{\max}_{c_i}(f)) \leftarrow$ PQ.pop() `/* Pop the element with the lowest` $\mathrm{h}^{\max}_{c_i}$`(f) */`
    for each $o \in \mathcal{O}, f \in pre(o)$ **do**
        $U(o) \leftarrow U(o) - 1$;
        if $U(o) = 0$ **then**
            `/*` $\mathrm{h}^{\max}_{c_i}$`(f) must be` $\max_{g \in \mathrm{pre}(o)} \mathrm{h}^{\max}_{c_i}(g)$ `because of PQ */`
            PQ.update($\{(g, \mathrm{h}^{\max}_{c_i}(f) + c_i(o)) \mid g \in \mathrm{add}(o)\}$);
            `/* PQ.update() can only decrease keys. */`
        end
    end
end

Justification Graph

Construction of Justification Graph for State s

- Given a set of operators $\mathcal{O}$, a cost function c_i and $\mathrm{h}^{\max}_{c_i}(\mathrm{f})$ for all $f \in \mathcal{F}$.
- Set $p(o) = \arg\max_{f \in \mathrm{pre}(o)} \mathrm{h}^{\max}_{c_i}(f)$ for each $o \in \mathcal{O}$ (breaking ties arbitrarily).
- Create a graph node for each fact $f \in \mathcal{F}$.
- For each operator o and each fact $f \in \mathrm{pre}(o)$, create an edge from $p(o)$ to f with cost $c_i(o)$ and label o.
- Add a node s' and zero-cost edges from s' to f for each $f \in s$.
- Add a node t and a zero-cost edge from $\arg\max_{f \in s_{goal}} \mathrm{h}^{\max}_{c_i}(f)$ to t.

s-t-cut

Construction of s-t-cut $C_i = (V_i^0, V_i^\star \cup V_i^b)$

- $V_i^\star$ contains all facts from which t can be reached through a zero-cost path.
- V_i^0 contains all facts reachable from s without passing through any fact from $V_i^\star$.
- V_i^b contains all remaining facts.
- Clearly $t \in V_i^\star$ and $s \in V_i^0$.
- Landmark L_i is a set of labels of the edges that cross the cut C_i (i.e., lead from V_i^0 to $V_i^\star$).

LM-Cut Heuristics

$h^{\mathrm{LM-cut}}$(s) for $\Pi = \langle \mathcal{F}, \mathcal{O}, s_{init}, s_{goal}, c \rangle$

1. Set $i = 1$ and $c_1 = c$ and $\mathrm{h}^{\mathrm{LM-cut}}(s) = 0$.
2. Compute $h^{\max}_{c_i}$(f) for all facts $f \in \mathcal{F}$, terminate if $\mathrm{h}^{\max}_{c_i}(s) = 0$.
3. Construct the justification graph.
4. Construct an s-t-cut C_i and obtain the landmark L_i.
5. Let $m_i = \min_{o \in L_i} c_i(o)$.
6. Define $c_{i+1}(o) = c_i(o)$ if $o \notin L_i$ and $c_{i+1}(o) = c_i(o) - m_i$ if $o \in L_i$.
7. Increase $\mathrm{h}^{\mathrm{LM-cut}}(s)$ by m_i.
8. Set $i = i + 1$ and continue with step 2.

Example

$\Pi = \langle \mathcal{F}, \mathcal{O}, s_{init}, s_{goal}, c \rangle$

- $\mathcal{F} = \{f_1, f_2, f_3, f_4, f_5, i, g\}$, $s_{init} = \{i\}$, $s_{goal} = \{g\}$
- $\mathcal{O}$:

op	pre	add	del	cost
o_1	$\{i\}$	$\{f_1, f_2\}$	$\emptyset$	2
o_2	$\{i\}$	$\{f_2, f_3\}$	$\emptyset$	3
o_3	$\{f_1, f_3\}$	$\{f_4\}$	$\{f_3\}$	1
o_4	$\{f_2, f_4\}$	$\{f_5\}$	$\{f_2\}$	3
o_5	$\{f_1, f_3, f_5\}$	$\{g\}$	$\{f_3, f_4\}$	1
o_6	$\{f_1\}$	$\{f_5\}$	$\{f_1, f_3\}$	5

Bibliography

[BH10] Blai Bonet and Malte Helmert.
Strengthening landmark heuristics via hitting sets.
In *19th European Conference on Artificial Intelligence, ECAI*, pages 329–334, 2010.

[HD09] Malte Helmert and Carmel Domshlak.
Landmarks, critical paths and abstractions: What's the difference anyway?
In *Proceedings of the 19th International Conference on Automated Planning and Scheduling (ICAPS)*, 2009.