

Planning and Optimization

E3. Landmarks: LM-Cut Heuristic

Malte Helmert and Gabriele Röger

Universität Basel

Planning and Optimization

— E3. Landmarks: LM-Cut Heuristic

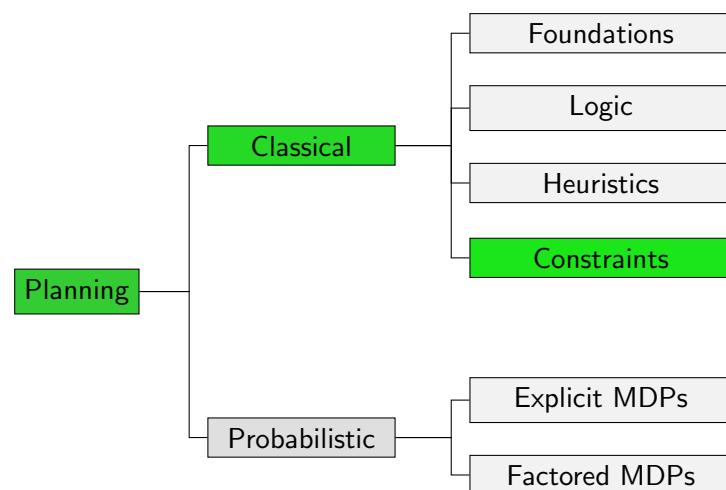
E3.1 i-g Form

E3.2 Cut Landmarks

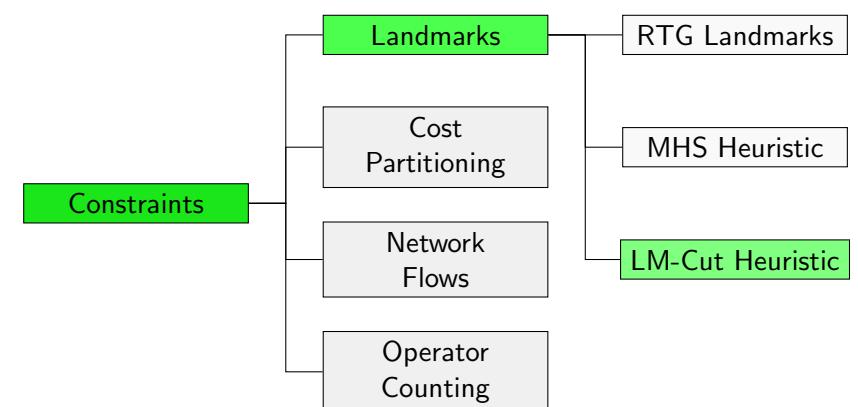
E3.3 The LM-Cut Heuristic

E3.4 Summary & Outlook

Content of this Course



Content of this Course: Constraints



Roadmap for this Chapter

- We first introduce a new **normal form** for **delete-free STRIPS tasks** that simplifies later definitions.
- We then present a method that **computes disjunctive action landmarks** for such tasks.
- We conclude with the **LM-cut heuristic** that builds on this method.

E3.1 i-g Form

Delete-Free STRIPS Planning Task in i-g Form (1)

In this chapter, we only consider **delete-free** STRIPS tasks in a special form:

Definition (i-g Form for Delete-free STRIPS)

A delete-free STRIPS planning task $\langle V, I, O, \gamma \rangle$ is in **i-g form** if

- V contains atoms i and g
- Initially exactly i is true: $I(v) = T$ iff $v = i$
- g is the only goal atom: $\gamma = \{g\}$
- Every action has at least one precondition.

Transformation to i-g Form

Every delete-free STRIPS task $\Pi = \langle V, I, O, \gamma \rangle$ can easily be transformed into an analogous task in i-g form.

- If i or g are in V already, rename them everywhere.
- Add i and g to V .
- Add an operator $\langle \{i\}, \{v \in V \mid I(v) = T\}, \{\}, 0 \rangle$.
- Add an operator $\langle \gamma, \{g\}, \{\}, 0 \rangle$.
- Replace all operator preconditions T with i .
- Replace initial state and goal.

For the remainder of this chapter, we assume tasks in i-g form.

Example: Delete-Free Planning Task in i-g Form

Example

Consider a delete-free STRIPS planning task $\langle V, I, O, \gamma \rangle$ with $V = \{i, a, b, c, d, g\}$, $I = \{i \mapsto T\} \cup \{v \mapsto F \mid v \in V \setminus \{i\}\}$, $\gamma = \{g\}$ and operators

- ▶ $o_{\text{blue}} = \langle \{i\}, \{a, b\}, \{\}, 4 \rangle$,
- ▶ $o_{\text{green}} = \langle \{i\}, \{a, c\}, \{\}, 5 \rangle$,
- ▶ $o_{\text{black}} = \langle \{i\}, \{b, c\}, \{\}, 3 \rangle$,
- ▶ $o_{\text{red}} = \langle \{b, c\}, \{d\}, \{\}, 2 \rangle$, and
- ▶ $o_{\text{orange}} = \langle \{a, d\}, \{g\}, \{\}, 0 \rangle$.

optimal solution to reach g from i :

- ▶ plan: $\langle o_{\text{blue}}, o_{\text{black}}, o_{\text{red}}, o_{\text{orange}} \rangle$
- ▶ cost: $4 + 3 + 2 + 0 = 9$ ($= h^+(I)$ because plan is optimal)

Justification Graphs

Definition (Precondition Choice Function)

A **precondition choice function** (pcf) $P : O \rightarrow V$ for a delete-free STRIPS task $\Pi = \langle V, I, O, \gamma \rangle$ in i-g form maps each operator to one of its preconditions (i.e. $P(o) \in \text{pre}(o)$ for all $o \in O$).

Definition (Justification Graphs)

Let P be a pcf for $\langle V, I, O, \gamma \rangle$ in i-g form. The **justification graph** for P is the directed, edge-labeled graph $J = \langle V, E \rangle$, where

- ▶ the vertices are the variables from V , and
- ▶ E contains an edge $P(o) \xrightarrow{o} a$ for each $o \in O$, $a \in \text{add}(o)$.

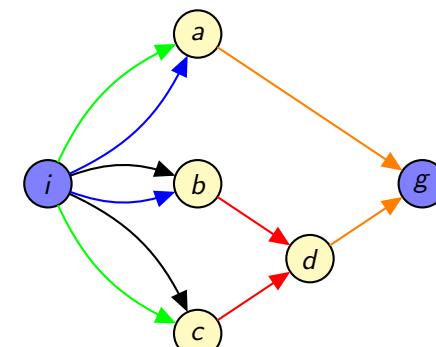
E3.2 Cut Landmarks

Justification Graphs

Example: Justification Graph

Example (Precondition Choice Function)

$P(o_{\text{blue}}) = P(o_{\text{green}}) = P(o_{\text{black}}) = i$, $P(o_{\text{red}}) = b$, $P(o_{\text{orange}}) = a$
 $P'(o_{\text{blue}}) = P'(o_{\text{green}}) = P'(o_{\text{black}}) = i$, $P'(o_{\text{red}}) = c$, $P'(o_{\text{orange}}) = d$

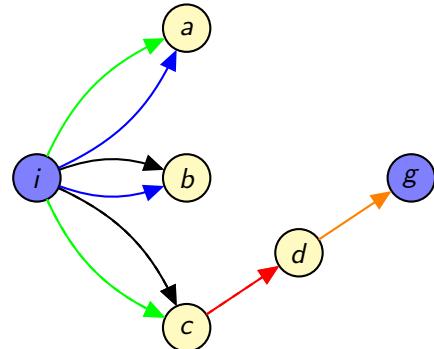


$o_{\text{blue}} = \langle \{i\}, \{a, b\}, \{\}, 4 \rangle$
 $o_{\text{green}} = \langle \{i\}, \{a, c\}, \{\}, 5 \rangle$
 $o_{\text{black}} = \langle \{i\}, \{b, c\}, \{\}, 3 \rangle$
 $o_{\text{red}} = \langle \{b, c\}, \{d\}, \{\}, 2 \rangle$
 $o_{\text{orange}} = \langle \{a, d\}, \{g\}, \{\}, 0 \rangle$

Cuts

Definition (Cut)

A **cut** in a justification graph is a subset C of its edges such that all paths from i to g contain an edge from C .



$$\begin{aligned} o_{\text{blue}} &= \langle \{i\}, \{a, b\}, \{\}, 4 \rangle \\ o_{\text{green}} &= \langle \{i\}, \{a, c\}, \{\}, 5 \rangle \\ o_{\text{black}} &= \langle \{i\}, \{b, c\}, \{\}, 3 \rangle \\ o_{\text{red}} &= \langle \{b, c\}, \{d\}, \{\}, 2 \rangle \\ o_{\text{orange}} &= \langle \{a, d\}, \{g\}, \{\}, 0 \rangle \end{aligned}$$

Cuts are Disjunctive Action Landmarks

Theorem (Cuts are Disjunctive Action Landmarks)

Let P be a pcf for $\langle V, I, O, \gamma \rangle$ (in i-g form) and C be a **cut** in the justification graph for P .

The set of **edge labels** from C (formally $\{o \mid \langle v, o, v' \rangle \in C\}$) is a **disjunctive action landmark** for I .

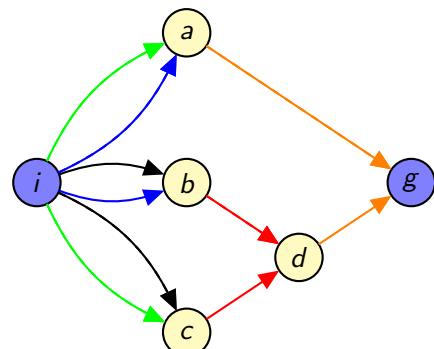
Proof idea:

- ▶ The justification graph corresponds to a simpler problem where some preconditions (those not picked by the pcf) are ignored.
- ▶ Cuts are landmarks for this simplified problem.
- ▶ Hence they are also landmarks for the original problem.

Example: Cuts in Justification Graphs

Example (Landmarks)

- ▶ $L_1 = \{o_{\text{orange}}\}$ (cost = 0)
- ▶ $L_2 = \{o_{\text{green}}, o_{\text{black}}\}$ (cost = 3)
- ▶ $L_3 = \{o_{\text{red}}\}$ (cost = 2)
- ▶ $L_4 = \{o_{\text{green}}, o_{\text{blue}}\}$ (cost = 4)



$$\begin{aligned} o_{\text{blue}} &= \langle \{i\}, \{a, b\}, \{\}, 4 \rangle \\ o_{\text{green}} &= \langle \{i\}, \{a, c\}, \{\}, 5 \rangle \\ o_{\text{black}} &= \langle \{i\}, \{b, c\}, \{\}, 3 \rangle \\ o_{\text{red}} &= \langle \{b, c\}, \{d\}, \{\}, 2 \rangle \\ o_{\text{orange}} &= \langle \{a, d\}, \{g\}, \{\}, 0 \rangle \end{aligned}$$

Power of Cuts in Justification Graphs

- ▶ Which landmarks can be computed with the cut method?
- ▶ **all interesting ones!**

Proposition (perfect hitting set heuristics)

Let \mathcal{L} be the set of **all** “cut landmarks” of a given planning task with initial state I . Then $h^{\text{MHS}}(\mathcal{L}) = h^+(I)$.

↝ Hitting set heuristic for \mathcal{L} is **perfect**.

Proof idea:

- ▶ Show 1:1 correspondence of hitting sets H for \mathcal{L} and plans, i.e., each hitting set for \mathcal{L} corresponds to a plan, and vice versa.

E3.3 The LM-Cut Heuristic

LM-Cut Heuristic

$h^{\text{LM-cut}}$: Helmert & Domshlak (2009)

Initialize $h^{\text{LM-cut}}(I) := 0$. Then iterate:

- 1 Compute h^{\max} values of the variables. Stop if $h^{\max}(g) = 0$.
- 2 Compute justification graph G for the P that chooses preconditions with maximal h^{\max} value
- 3 Determine the goal zone V_g of G that consists of all nodes that have a zero-cost path to g .
- 4 Compute the cut L that contains the labels of all edges $\langle v, o, v' \rangle$ such that $v \notin V_g$, $v' \in V_g$ and v can be reached from i without traversing a node in V_g . It is guaranteed that $\text{cost}(L) > 0$.
- 5 Increase $h^{\text{LM-cut}}(I)$ by $\text{cost}(L)$.
- 6 Decrease $\text{cost}(o)$ by $\text{cost}(L)$ for all $o \in L$.

LM-Cut Heuristic: Motivation

- ▶ In general, there are exponentially many pcfs, hence computing all relevant landmarks is not tractable.
- ▶ The **LM-cut heuristic** is a method that chooses pcfs and computes cuts in a **goal-oriented** way.
- ▶ As a side effect, it computes
 - ▶ a cost partitioning over multiple instances of h^{\max} that is also
 - ▶ a **saturated cost partitioning** over disjunctive action landmarks.
- ~~ currently one of the best admissible planning heuristic

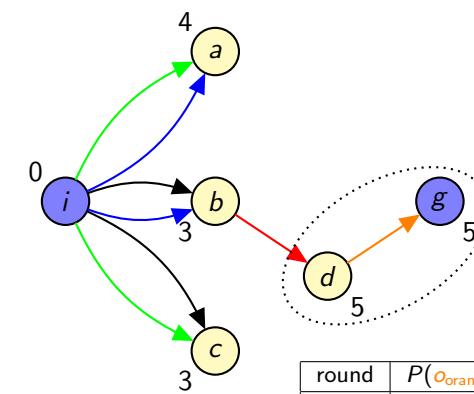
LM-Cut Heuristic

$h^{\text{LM-cut}}$: Helmert & Domshlak (2009)

Initialize $h^{\text{LM-cut}}(I) := 0$. Then iterate:

- 1 Compute h^{\max} values of the variables. Stop if $h^{\max}(g) = 0$.
- 2 Compute justification graph G for the P that chooses preconditions with maximal h^{\max} value
- 3 Determine the goal zone V_g of G that consists of all nodes that have a zero-cost path to g .
- 4 Compute the cut L that contains the labels of all edges $\langle v, o, v' \rangle$ such that $v \notin V_g$, $v' \in V_g$ and v can be reached from i without traversing a node in V_g . It is guaranteed that $\text{cost}(L) > 0$.
- 5 Increase $h^{\text{LM-cut}}(I)$ by $\text{cost}(L)$.
- 6 Decrease $\text{cost}(o)$ by $\text{cost}(L)$ for all $o \in L$.

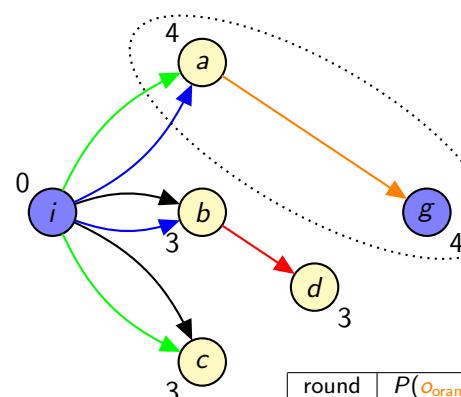
Example: Computation of LM-Cut



$o_{\text{blue}} = \langle \{i\}, \{a, b\}, \{\}, 4 \rangle$
 $o_{\text{green}} = \langle \{i\}, \{a, c\}, \{\}, 5 \rangle$
 $o_{\text{black}} = \langle \{i\}, \{b, c\}, \{\}, 3 \rangle$
 $o_{\text{red}} = \langle \{b, c\}, \{d\}, \{\}, 0 \rangle$
 $o_{\text{orange}} = \langle \{a, d\}, \{g\}, \{\}, 0 \rangle$

round	$P(o_{\text{orange}})$	$P(o_{\text{red}})$	landmark	cost
1	d	b	$\{o_{\text{red}}\}$	2
				$h^{\text{LM-cut}}(I)$
				2

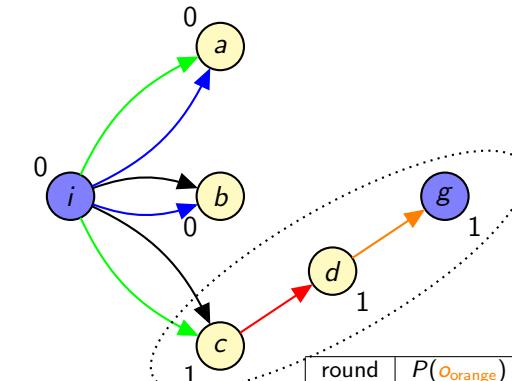
Example: Computation of LM-Cut



$o_{blue} = \langle \{i\}, \{a, b\}, \{\}, 0 \rangle$
 $o_{green} = \langle \{i\}, \{a, c\}, \{\}, 1 \rangle$
 $o_{black} = \langle \{i\}, \{b, c\}, \{\}, 3 \rangle$
 $o_{red} = \langle \{b, c\}, \{d\}, \{\}, 0 \rangle$
 $o_{orange} = \langle \{a, d\}, \{g\}, \{\}, 0 \rangle$

round	$P(o_{orange})$	$P(o_{red})$	landmark	cost
1	d	b	$\{o_{red}\}$	2
2	a	b	$\{o_{green}, o_{blue}\}$	4
$h^{\text{LM-cut}}(I)$				6

Example: Computation of LM-Cut



$o_{blue} = \langle \{i\}, \{a, b\}, \{\}, 0 \rangle$
 $o_{green} = \langle \{i\}, \{a, c\}, \{\}, 0 \rangle$
 $o_{black} = \langle \{i\}, \{b, c\}, \{\}, 2 \rangle$
 $o_{red} = \langle \{b, c\}, \{d\}, \{\}, 0 \rangle$
 $o_{orange} = \langle \{a, d\}, \{g\}, \{\}, 0 \rangle$

round	$P(o_{orange})$	$P(o_{red})$	landmark	cost
1	d	b	$\{o_{red}\}$	2
2	a	b	$\{o_{green}, o_{blue}\}$	4
3	d	c	$\{o_{green}, o_{black}\}$	1
$h^{\text{LM-cut}}(I)$				7

Properties of LM-Cut Heuristic

Theorem

Let $\langle V, I, O, \gamma \rangle$ be a delete-free STRIPS task in i-g normal form.
The LM-cut heuristic is admissible: $h^{\text{LM-cut}}(I) \leq h^*(I)$.

Proof omitted.

If Π is not delete-free, we can compute $h^{\text{LM-cut}}$ on Π^+ .
Then $h^{\text{LM-cut}}$ is bounded by h^+ .

E3.4 Summary & Outlook

Summary

- ▶ Cuts in justification graphs are a general method to find disjunctive action landmarks.
- ▶ The minimum hitting set over all cut landmarks is a perfect heuristic for delete-free planning tasks.
- ▶ The LM-cut heuristic is an admissible heuristic based on these ideas.

Literature (1)

References on landmark heuristics:

- ▶ Julie Porteous, Laura Sebastian and Joerg Hoffmann. On the Extraction, Ordering, and Usage of Landmarks in Planning. *Proc. ECP 2001*, pp. 174–182, 2013. Introduces landmarks.
- ▶ Malte Helmert and Carmel Domshlak. Landmarks, Critical Paths and Abstractions: What's the Difference Anyway? *Proc. ICAPS 2009*, pp. 162–169, 2009. Introduces cut landmarks and LM-cut heuristic.

Literature (2)

- ▶ Lin Zhu and Robert Givan. Landmark Extraction via Planning Graph Propagation. *Doctoral Consortium ICAPS 2003*, 2003. Core idea for complete landmark generation.
- ▶ Emil Keyder, Silvia Richter and Malte Helmert. Sound and Complete Landmarks for And/Or Graphs *Proc. ECAI 2010*, pp. 335–340, 2010. Introduces landmarks from AND/OR graphs and usage of Π^m compilation.

Literature (3)

- ▶ Silvia Richter and Matthias Westphal. The LAMA Planner: Guiding Cost-Based Anytime Planning with Landmarks. *JAIR 39 (2010)*, pp. 127–177, 2010. Introduces landmark-count heuristic and contains another landmark generation method.
- ▶ Erez Karpas and Carmel Domshlak. Cost-Optimal Planning with Landmarks. *Proc. IJCAI 2009*, pp. 1728–1733, 2009. Introduces admissible variant of landmark heuristic.