Strengthening Landmark Heuristics via Hitting Sets

Blai Bonet\textsuperscript{1}  Malte Helmert\textsuperscript{2}

\textsuperscript{1}Universidad Simón Bolívar, Caracas, Venezuela

\textsuperscript{2}Albert-Ludwigs-Universität Freiburg, Germany

ECAI 2010 – August 18th, 2010
Area: **heuristics** for optimal classical planning

**Our contribution**
- **stronger** way of exploiting landmarks for heuristic functions
- **systematic** way of generating landmarks for delete relaxation
- theoretical results relating new ideas to
  - admissible landmark heuristics (Karpas & Domshlak, 2009)
  - landmark-cut heuristic (Helmert & Domshlak, 2009)
  - optimal delete relaxation $h^+$ (Hoffmann & Nebel, 2001)
  - fixed-parameter tractability of problems of hitting sets
- new poly-time heuristic family that dominates landmark-cut
<table>
<thead>
<tr>
<th>Relaxed planning</th>
<th>Landmarks</th>
<th>Exploiting LMs</th>
<th>Generating LMs</th>
<th>Improved LM-cut</th>
<th>Conclusion</th>
</tr>
</thead>
</table>

Relaxed planning
Optimal planning:

- shortest paths in huge implicit graphs
- no formal definition here

What we need to know:

- state-of-the-art planners: heuristic search
- optimal planners: A* + heuristics
- many use delete relaxation ("relaxed planning tasks")
- want accurate estimates of optimal delete relaxation cost $h^+$
Relaxed planning tasks

Obtained by removing the deletes of each action

**Definition (relaxed planning task)**

\( F \): finite set of facts

- **initial facts** \( I \subseteq F \) are given
- **goal facts** \( G \subseteq F \) must be reached
- **operators** of the form \( o[4] : a, b \rightarrow c, d \)

  read: If we already have facts \( a \) and \( b \) (preconditions \( pre(o) \)),
  we can apply \( o \), paying 4 units (cost \( cost(o) \)),
  to obtain facts \( c \) and \( d \) (effects \( eff(o) \))

For simplicity (WLOG): assume \( I = \{i\} \), \( G = \{g\} \), all \( pre(o) \neq \emptyset \)
Example: relaxed planning task

Example

\[ o_1[3] : i \rightarrow a, b \]
\[ o_2[4] : i \rightarrow a, c \]
\[ o_3[5] : i \rightarrow b, c \]
\[ o_4[0] : a, b, c \rightarrow g \]

One way to reach \( \{g\} \) from \( \{i\} \):

- apply sequence \( o_1, o_2, o_4 \) (plan)
- cost: \( 3 + 4 + 0 = 7 \) (optimal)
Optimal relaxed cost

- \( h^+(I) \) : minimal total cost to reach \( G \) from \( I \)
- **Very good heuristic** function for optimal planning
- **NP-hard** to compute (Bylander, 1994)
  or approximate by constant factor (Betz & Helmert, 2009)

\[ \Rightarrow \] use polynomial-time **admissible heuristics**
Landmarks
The most accurate current heuristics are based on landmarks.

**Definition (landmark)**

A (disjunctive action) **landmark** is a set of operators $L$ such that each plan must contain some element of $L$.

The **cost** of a landmark, $cost(L)$, is $\min_{o \in L} cost(o)$.

$\Rightarrow$ the cost of any landmark is a (crude) admissible heuristic
Example: landmarks

Example

\[
\begin{align*}
o_1[3] & : i \rightarrow a, b \\
o_2[4] & : i \rightarrow a, c \\
o_3[5] & : i \rightarrow b, c \\
o_4[0] & : a, b, c \rightarrow g
\end{align*}
\]

Some landmarks:

- \( W = \{o_4\} \) (cost 0)
- \( X = \{o_1, o_2\} \) (cost 3)
- \( Y = \{o_1, o_3\} \) (cost 3)
- \( Z = \{o_2, o_3\} \) (cost 4)
- but also: \( \{o_1, o_2, o_3\} \) (cost 3), \( \{o_1, o_2, o_4\} \) (cost 0), \ldots \)
Exploiting landmarks
Exploiting landmarks

Assume we are given landmark set \( \mathcal{L} = \{W, X, Y, Z\} \) (later: how to find such landmarks)

How do we exploit \( \mathcal{L} \) for heuristics?

- **sum** of costs \( 0 + 3 + 3 + 4 = 10 \) \( \Rightarrow \) inadmissible!
- **maximum** of costs: \( \max \{0, 3, 3, 4\} = 4 \) \( \Rightarrow \) weak
- best previous approach: optimal cost partitioning
Optimal cost partitioning (Karpas & Domshlak (2009))

Example

\[ cost(o_1) = 3, \quad cost(o_2) = 4, \quad cost(o_3) = 5, \quad cost(o_4) = 0 \]

\[ L = \{W, X, Y, Z\} \]
with \( W = \{o_4\}, \quad X = \{o_1, o_2\}, \quad Y = \{o_1, o_3\}, \quad Z = \{o_2, o_3\} \)

LP: maximize \( w + x + y + z \) subject to \( w, x, y, z \geq 0 \) and

\[
\begin{align*}
x + y & \leq 3 & o_1 \\
x + z & \leq 4 & o_2 \\
y + z & \leq 5 & o_3 \\
w & \leq 0 & o_4 \\
\end{align*}
\]

solution: \( w = 0, \quad x = 1, \quad y = 2, \quad z = 3 \) \( \Rightarrow h^L(I) = 6 \)
Hitting sets

**Definition (hitting set)**

Given: finite set $A$, subset family $\mathcal{F} \subseteq 2^A$, costs $c : A \rightarrow \mathbb{R}_0^+$

Hitting set:
- subset $H \subseteq A$ that “hits” all subsets in $\mathcal{F}$:
  \[ H \cap S \neq \emptyset \text{ for all } S \in \mathcal{F} \]
- cost of $H$: $\sum_{a \in H} c(a)$

**Minimum hitting set (MHS):**
- minimizes cost
- classical NP-complete problem (Karp, 1972)
Can view landmark sets (with operator costs) as instances of minimum hitting set problem

**Example**

\[ A = \{ o_1, o_2, o_3, o_4 \} \]

\[ \mathcal{F} = \{ W, X, Y, Z \} \]

with \( W = \{ o_4 \}, \ X = \{ o_1, o_2 \}, \ Y = \{ o_1, o_3 \}, \ Z = \{ o_2, o_3 \} \)

\[ c(o_1) = 3, \quad c(o_2) = 4, \quad c(o_3) = 5, \quad c(o_4) = 0 \]

Minimum hitting set: \( \{ o_1, o_2, o_4 \} \) with cost \( 3 + 4 + 0 = 7 \)
Let $\mathcal{L}$ be a set of landmarks.

**Theorem (hitting set heuristics are admissible)**

Let $h_{\text{MHS}}(I)$ be the minimum hitting set cost for $\langle O, \mathcal{L}, \text{cost} \rangle$.

Then:

1. $h_{\text{MHS}}(I) \geq h^L(I)$ (hitting sets dominate cost partitioning)
2. $h_{\text{MHS}}(I) \leq h^+(I)$ (hitting set heuristics are admissible)
Generating landmarks
How do we **generate** landmarks in the first place?

- most successful previous approach: **LM-cut procedure** (Helmert & Domshlak, 2009)

- we present a generalization based on:
  - construction of **justification graph**
  - extraction of landmarks from justification graph
Justification graphs

**Definition (precondition choice function)**

A **precondition choice function** (pcf) \( D : O \rightarrow F \) maps each operator to one of its preconditions.

**Definition (justification graph)**

The **justification graph** for pcf \( D \) is an arc-labeled digraph with

- **vertices**: the facts \( F \)
- **arcs**: \( D(o) \xrightarrow{o} e \) for each operator \( o \) and effect \( e \in \text{eff}(o) \)
Example: justification graph

Example

\[
\text{pcf } D: \quad D(o_1) = D(o_2) = D(o_3) = i, \quad D(o_4) = a
\]

\[
\begin{align*}
o_1[3] : & \quad i \rightarrow a, b \\
o_2[4] : & \quad i \rightarrow a, c \\
o_3[5] : & \quad i \rightarrow b, c \\
o_4[0] : & \quad a, b, c \rightarrow g
\end{align*}
\]
Example: cuts of a justification graph

Example

Landmark $W = \{o_4\}$ (cost 0)

\[
\begin{align*}
o_1[3] : i &\rightarrow a, b \\
o_2[4] : i &\rightarrow a, c \\
o_3[5] : i &\rightarrow b, c \\
o_4[0] : a, b, c &\rightarrow g
\end{align*}
\]
Example: cuts of a justification graph

**Example**

Landmark $X = \{o_1, o_2\}$ (cost 3)

- $o_1[3] : i \rightarrow a, b$
- $o_2[4] : i \rightarrow a, c$
- $o_3[5] : i \rightarrow b, c$
- $o_4[0] : a, b, c \rightarrow g$
Example: cuts of a justification graph

Example

Landmark $Y = \{o_1, o_3\}$ (cost 3)

- $o_1[3] : i \rightarrow a, b$
- $o_2[4] : i \rightarrow a, c$
- $o_3[5] : i \rightarrow b, c$  
- $o_4[0] : a, b, c \rightarrow g$
Example: cuts of a justification graph

Example

Landmark $Z = \{o_2, o_3\}$ (cost 4)

- $o_1[3]: i \rightarrow a, b$
- $o_2[4]: i \rightarrow a, c$
- $o_3[5]: i \rightarrow b, c$
- $o_4[0]: a, b, c \rightarrow g$
Power of justification graph cuts

- Which landmarks can be generated with the cut method?
- All interesting ones!

**Theorem (perfect hitting set heuristics)**

Let $\mathcal{L}$ be the set of all “cut landmarks”.

Then $h^{MHS}(I) = h^+(I)$.

$\Rightarrow$ hitting set heuristic over $\mathcal{L}$ is perfect
<table>
<thead>
<tr>
<th>Relaxed planning</th>
<th>Landmarks</th>
<th>Exploiting LMs</th>
<th>Generating LMs</th>
<th>Improved LM-cut</th>
<th>Conclusion</th>
</tr>
</thead>
</table>

**Improving the LM-cut heuristic**
Polynomial hitting set heuristics

How practical are our results?

- minimum hitting set is **NP-hard**
- number of cut landmarks is **exponential**

We show how to apply our results to derive

- **polynomial** heuristics which
- dominate the **LM-cut heuristic**
LM-cut heuristic

- Computes a collection of landmarks by using pcfs that choose preconditions maximizing $h^{\text{max}}$
- Derived landmarks are pairwise disjoint
- Thus, costs can be combined (admissibly) with addition
Improved LM-cut

Improve the LM-cut heuristic by

1. Generating more landmarks:
   - Perform the LM-cut computation \( p \) times (parameter)
   - Use random tie-breaking to make runs different
   - Collect all generated landmarks in a set \( L \).

2. Exploiting them in a smarter way:
   - Introduce a width parameter \( k \) for hitting set instances such that MHS is fixed-parameter tractable w.r.t. \( k \)
   - Remove some landmarks from \( L \) to bound the width
   - Solve resulting MHS problem in polynomial time
## Preliminary experiments

<table>
<thead>
<tr>
<th>#</th>
<th>LM-cut</th>
<th>$h_{p,k}^{LM}$ with $k = 5$</th>
<th>$h_{p,k}^{LM}$ with $k = 10$</th>
<th>$h_{p,k}^{LM}$ with $k = 15$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>$p = 3$</td>
<td>$p = 4$</td>
<td>$p = 5$</td>
</tr>
<tr>
<td>Pipeworl-NoTankage (rel. error of LM-cut wrt $h^+ = 19.45%$)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>06</td>
<td>107</td>
<td>45.8</td>
<td>54.2</td>
<td>67.3</td>
</tr>
<tr>
<td>07</td>
<td>3</td>
<td>100.0</td>
<td>100.0</td>
<td>100.0</td>
</tr>
<tr>
<td>08</td>
<td>84</td>
<td>47.6</td>
<td>57.1</td>
<td>81.0</td>
</tr>
<tr>
<td>10</td>
<td>137,092</td>
<td>30.2</td>
<td>40.1</td>
<td>46.9</td>
</tr>
<tr>
<td>Pipeworl-Tankage (rel. error of LM-cut wrt $h^+ = 18.42%$)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>05</td>
<td>74</td>
<td>58.1</td>
<td>70.3</td>
<td>70.3</td>
</tr>
<tr>
<td>06</td>
<td>223</td>
<td>41.7</td>
<td>52.0</td>
<td>60.5</td>
</tr>
<tr>
<td>07</td>
<td>323</td>
<td>100.0</td>
<td>100.0</td>
<td>100.0</td>
</tr>
<tr>
<td>08</td>
<td>36,203</td>
<td>77.3</td>
<td>84.9</td>
<td>87.6</td>
</tr>
<tr>
<td>Openstacks (rel. error of LM-cut wrt $h^+ = 18.09%$)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>04</td>
<td>1,195</td>
<td>53.4</td>
<td>57.8</td>
<td>59.0</td>
</tr>
<tr>
<td>05</td>
<td>1,195</td>
<td>52.6</td>
<td>57.4</td>
<td>59.7</td>
</tr>
<tr>
<td>06</td>
<td>211,175</td>
<td>64.6</td>
<td>64.9</td>
<td>65.2</td>
</tr>
<tr>
<td>07</td>
<td>266,865</td>
<td>60.7</td>
<td>61.3</td>
<td>61.8</td>
</tr>
<tr>
<td>Freecell (rel. error of LM-cut wrt $h^+ = 13.92%$)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>pf4</td>
<td>36,603</td>
<td>70.7</td>
<td>75.2</td>
<td>78.4</td>
</tr>
<tr>
<td>pf5</td>
<td>53,670</td>
<td>73.6</td>
<td>76.0</td>
<td>77.9</td>
</tr>
<tr>
<td>2-5</td>
<td>277</td>
<td>72.9</td>
<td>73.3</td>
<td>74.0</td>
</tr>
<tr>
<td>3-4</td>
<td>17,763</td>
<td>44.6</td>
<td>62.8</td>
<td>73.1</td>
</tr>
</tbody>
</table>
Conclusion
Summary:

- **Hitting sets** for landmarks are more informative than optimal cost partitioning.

- **Cuts** in justification graphs offer a *principled* and *complete* method for generating landmarks.

- Hitting sets over **all cut landmarks** are perfect heuristics for delete relaxations.

- These concepts can be exploited in *practical heuristics*.