Landmarks Revisited

Silvia Richter\textsuperscript{1} \hspace{1em} Malte Helmert\textsuperscript{2} \hspace{1em} Matthias Westphal\textsuperscript{2}

\textsuperscript{1}Griffith University \& NICTA, Australia

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

AAAI 2008
Outline

1. Introduction to $SAS^+$ Planning
2. Landmarks in Previous Work
3. Using Landmarks as Pseudo-Heuristic
4. Extended Landmark Generation
SAS$^+$ planning task: $$\Pi = \langle \mathcal{V}, \mathcal{A}, s_0, s_\star \rangle$$

- $\mathcal{V}$: state variables with finite domain $\mathcal{D}_v$
  - Fact: variable-value pair $v \mapsto d$ ($v \in \mathcal{V}$, $d \in \mathcal{D}_v$)
  - State: variable assignment for all $v \in \mathcal{V}$
- $\mathcal{A}$: actions $\langle \text{pre}, \text{eff} \rangle$, with $\text{pre}$, $\text{eff}$ fact sets
  - Action $a = \langle \text{pre}, \text{eff} \rangle$ applicable in state $s$ if $\text{pre} \subseteq s$
  - Applying $a$ in $s$ updates $s$
- $s_0$: initial state
- $s_\star$: partial variable assignment called the goal

Sequence of actions $\pi$ a plan iff $s_\star \subseteq s_0[\pi]$. 
Encoding of example task

\[
\mathcal{V} = \{v_o, v_t, v_p\} \\
\mathcal{D}_{v_o} = \{A, B, C, D, E, t, p\} \quad \mathcal{D}_{v_t} = \{A, B, C, D\}, \quad \mathcal{D}_{v_p} = \{C, E\} \\
\mathcal{A} = \{\text{drive-t-D-B, load-o-t-B, ...}\} \\
\mathcal{s}_0 = \{v_o \mapsto B, \quad v_t \mapsto D, \quad v_p \mapsto E\} \\
\mathcal{s}_* = \{v_o \mapsto E\}
\]
Encoding of example task

\[\mathcal{V} = \{v_o, v_t, v_p\}\]

\[\mathcal{D}_{v_o} = \{A, B, C, D, E, t, p\} \quad \mathcal{D}_{v_t} = \{A, B, C, D\}, \quad \mathcal{D}_{v_p} = \{C, E\}\]

\[\mathcal{A} = \{\text{drive-t-D-B, load-o-t-B, \ldots}\}\]

\[s_0 = \{v_o \mapsto B, \quad v_t \mapsto D, \quad v_p \mapsto E\}\]

\[s_\star = \{v_o \mapsto E\}\]
Encoding of example task

\[ \mathcal{V} = \{ \mathcal{V}_o, \mathcal{V}_t, \mathcal{V}_p \} \]

\[ D_{\mathcal{V}_o} = \{ A, B, C, D, E, t, p \} \]

\[ D_{\mathcal{V}_t} = \{ A, B, C, D \}, \quad D_{\mathcal{V}_p} = \{ C, E \} \]

\[ \mathcal{A} = \{ \text{drive-t-D-B, load-o-t-B, } \ldots \} \]

\[ s_0 = \{ \mathcal{V}_o \mapsto B, \mathcal{V}_t \mapsto D, \mathcal{V}_p \mapsto E \} \]

\[ s_* = \{ \mathcal{V}_o \mapsto E \} \]
Introduction to SAS$^+$ Planning
Landmarks in Previous Work
Using Landmarks as Pseudo-Heuristic
Extended Landmark Generation

Encoding of example task

\[ V = \{v_o, v_t, v_p\} \]
\[ D_{v_o} = \{A, B, C, D, E, t, p\} \quad D_{v_t} = \{A, B, C, D\}, \quad D_{v_p} = \{C, E\} \]
\[ A = \{\text{drive-t-D-B, load-o-t-B, ...}\} \]
\[ s_0 = \{v_o \mapsto B, \quad v_t \mapsto D, \quad v_p \mapsto E\} \]
\[ s_* = \{v_o \mapsto E\} \]
Encoding of example task

\[ \mathcal{V} = \{ v_o, v_t, v_p \} \]
\[ D_{v_o} = \{ A, B, C, D, E, t, p \} \quad D_{v_t} = \{ A, B, C, D \}, \quad D_{v_p} = \{ C, E \} \]
\[ \mathcal{A} = \{ \text{drive-t-D-B}, \quad \text{load-o-t-B}, \ldots \} \]
\[ s_0 = \{ v_o \mapsto B, \quad v_t \mapsto D, \quad v_p \mapsto E \} \]
\[ s_\star = \{ v_o \mapsto E \} \]
Encoding of example task

\( \mathcal{V} = \{ \nu_o, \nu_t, \nu_p \} \)

\( \mathcal{D}_{\nu_o} = \{ A, B, C, D, E, t, p \} \)

\( \mathcal{D}_{\nu_t} = \{ A, B, C, D \} \)

\( \mathcal{D}_{\nu_p} = \{ C, E \} \)

\( \mathcal{A} = \{ \text{drive-t-D-B, load-o-t-B, ...} \} \)

\( s_0 = \{ \nu_o \mapsto B, \nu_t \mapsto D, \nu_p \mapsto E \} \)

\( s_\star = \{ \nu_o \mapsto E \} \)

\( o \text{-at-E} \)
load-o-t-B : \langle \text{Pre} = \{v_o \mapsto B, \ v_t \mapsto B\}, \text{Eff} = \{v_o \mapsto t\} \rangle
Introduction to SAS^+ Planning
Landmarks in Previous Work
Using Landmarks as Pseudo-Heuristic
Extended Landmark Generation

Preferred Operators

- Improvement of heuristic search approaches (Helmert 2006)
- Idea: prefer actions that are likely to improve heuristic value
- E.g. those which are part of plan for simplified problem
Landmarks in Previous Work

- Facts that **must** be true in every plan  
  (Porteous et al. 2001 & 2002; Hoffmann et al. 2004)
- Intuitively helpful to direct search
- Automatically found, incl. orderings
Landmarks in Previous Work cont’d

Find landmarks by backchaining

- Every goal is a landmark
- If B is landmark and all actions that first achieve B have A as precondition, then A is a landmark
- Approximation with RPGs: consider all achievers “possibly before” B (Porteous et al. 2002)

- Disjunctive landmarks also possible: \((o-in-p_1 \lor o-in-p_2)\)
Use as **subgoals**, then simply concatenate plans of subtasks ("LM-local")

- Greatly speeds up search in many domains
- But: bad-quality plans, incomplete (dead ends)
- Any base planner possible for subtasks
Outline

1. Introduction to SAS+ Planning
2. Landmarks in Previous Work
3. Using Landmarks as Pseudo-Heuristic
4. Extended Landmark Generation
Novel usage of landmarks

- **Pseudo-Heuristic** = $\#$ landmarks that still need to be achieved
- Take orderings into account (see paper for details)
- **Preferred operators** = landmark-achieving operators or operators in relaxed plan to nearest landmark
- Combination with other heuristics through multi-heuristic BFS (Helmert 2006)

Experiments with several heuristics (FF, CG, blind) on all tasks from past planning competitions
Results: % Tasks solved (Average)

<table>
<thead>
<tr>
<th>Base Heuristic</th>
<th>Algorithm</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>base</td>
<td>LM-local</td>
</tr>
<tr>
<td>FF heuristic</td>
<td>87</td>
<td>82</td>
</tr>
<tr>
<td>CG heuristic</td>
<td>74</td>
<td>66</td>
</tr>
<tr>
<td>blind heuristic</td>
<td>25</td>
<td>52</td>
</tr>
</tbody>
</table>

Note: updated results for LM-local

- With all 3 heuristics, LM-heur dominates other approaches
- LM-local worse than base with CG and blind heuristic (dead ends in 8 domains)
- FF-heuristic: base and LM-local are close.
Results: %Tasks solved (Average)

<table>
<thead>
<tr>
<th>Base Heuristic</th>
<th>Algorithm base</th>
<th>LM-local</th>
<th>LM-heur</th>
</tr>
</thead>
<tbody>
<tr>
<td>FF heuristic</td>
<td>87</td>
<td>82</td>
<td>88</td>
</tr>
<tr>
<td>CG heuristic</td>
<td>74</td>
<td>66</td>
<td>87</td>
</tr>
<tr>
<td>blind heuristic</td>
<td>25</td>
<td>52</td>
<td>84</td>
</tr>
</tbody>
</table>

- With all 3 heuristics, LM-heur dominates other approaches
- LM-local worse than base with CG and blind heuristic (dead ends in 8 domains)
- FF-heuristic: base and LM-local are close...
### Results: #Tasks solved exclusively (FF heuristic)

<table>
<thead>
<tr>
<th>Domain</th>
<th>FF heuristic base</th>
<th>LM-heur</th>
</tr>
</thead>
<tbody>
<tr>
<td>Airport (50)</td>
<td>6</td>
<td>2</td>
</tr>
<tr>
<td>Depot (22)</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>Freecell (80)</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>Logistics-1998 (35)</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>Miconic-FullADL (150)</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>MPrime (35)</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>Mystery (30)</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>Pathways (30)</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>Philosophers (48)</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>Pipesworld-NoTankage (50)</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>Pipesworld-Tankage (50)</td>
<td>1</td>
<td>5</td>
</tr>
<tr>
<td>Schedule (150)</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>Storage (30)</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>12</strong></td>
<td><strong>25</strong></td>
</tr>
</tbody>
</table>

LM-heur solves twice as many tasks exclusively as base
Outline

1. Introduction to SAS+ Planning
2. Landmarks in Previous Work
3. Using Landmarks as Pseudo-Heuristic
4. Extended Landmark Generation
Extended Landmark Generation

- Adapted previous procedures to $\text{SAS}^+$ planning
- Admit disjunctive landmarks
- Find additional landmarks through DTGs
The **domain transition graph** of $v \in \mathcal{V}$ (DTG$_v$) represents how the value of $v$ can change.

Given: a SAS$^+$ task $\langle \mathcal{V}, \mathcal{A}, s_0, s_\star \rangle$

DTG$_v$ is a directed graph with nodes $D_v$ that has arc $\langle d, d' \rangle$ iff

- $d \neq d'$, and
- $\exists$ action with $v \leftarrow d'$ as effect, and either
  - $v \leftarrow d$ as precondition, or
  - no precondition on $v$
DTG Example

DTG for $v_o$:
DTG Example

**DTG for \( v_o \):**

![Diagram](attachment:image.png)
DTG Example

DTG for $v_o$:

- B to t
- load-o-t-B

Diagram showing nodes A, B, C, D, and E with connections between them.
Extended Landmark Generation

- Find additional landmarks through DTGs: if
  - $s_0(v) = d_0$,
  - $v \mapsto d$ landmark, and
  - every path from $d_0$ to $d$ passes through $d'$,

  then $v \mapsto d'$ landmark

- No further improvement in % solved, but shorter plans
Find additional landmarks through DTGs: if
- \( s_0(v) = d_0 \),
- \( v \mapsto d \) landmark, and
- every path from \( d_0 \) to \( d \) passes through \( d' \),
then \( v \mapsto d' \) landmark

No further improvement in % solved, but shorter plans
Extended landmark generation – Plan length

- Local: plans 6% longer than with base
- Heur: plans 1% shorter
- Heur with extended LMs: plans 3% shorter

![Plan length in Schedule](image1)

![Plan length in Gripper](image2)
Remarks on Runtime

- LM generation usually < 1 sec. (max. 2 min.)
- During search: slight overhead through landmarks (≤ 18%)
- Overhead typically outweighed by benefit in larger problems
Summary

- Landmark heuristic significantly improves existing heuristics
- More tasks solved
- Better quality of solutions (plan lengths)
- Complete, unlike previous local search approach
- First approach that handles disjunctive landmarks
Summary

- Landmark heuristic significantly improves existing heuristics
- More tasks solved
- Better quality of solutions (plan lengths)
- Complete, unlike previous local search approach
- First approach that handles disjunctive landmarks

Thank you!