Focusing on What Really Matters: Irrelevance Pruning in M&S

Álvaro Torralba, Peter Kissmann
Saarland University, Germany

SoCS 2015, June 11
Session with ICAPS 2015
Motivation: Irrelevance Pruning

- Last Tuesday: $h^2$-based preprocessor
  - Simplify the task in a preprocessing step
  - Remove operators that cannot possibly belong to any plan
  - Very useful!!!!

- Today: Can we simplify the tasks even further?
Motivation: Irrelevance Pruning

- Last Tuesday: $h^2$-based preprocessor
  - Simplify the task in a preprocessing step
  - Remove operators that cannot possibly belong to any plan
  - Very useful!!!!!
Motivation: Irrelevance Pruning

- Last Tuesday: \( h^2 \)-based preprocessor
  - Simplify the task in a preprocessing step
  - Remove operators that cannot possibly belong to any plan
  - Very useful!!!!

- Today: Can we simplify the tasks even further?

Diagram:

- (Truck)
- (Package)
Merge-and-Shrink Heuristic

An admissible abstraction heuristic for cost-optimal planning

1. Start with the projection over variables: $v_1, v_2, v_3, v_4$
2. Merge: replace $\Theta_i$ and $\Theta_j$ by their product
3. Shrink: replace $\Theta_i$ by its abstraction $\alpha(\Theta_i)$

\[ \text{L} \]

\[ \Theta_1 \quad \Theta_2 \quad \Theta_3 \quad \Theta_4 \]
Merge-and-Shrink Heuristic

An admissible abstraction heuristic for cost-optimal planning

1. Start with the projection over variables: \( \nu_1, \nu_2, \nu_3, \nu_4 \)
2. Merge: replace \( \Theta_i \) and \( \Theta_j \) by their product
3. Shrink: replace \( \Theta_i \) by its abstraction \( \alpha(\Theta_i) \)

\[
\begin{array}{c}
\Theta_1 \otimes \Theta_2 \\
\Theta_3 \\
\Theta_4 \\
\end{array}
\]

\( \mathcal{L} \)
Merge-and-Shrink Heuristic

An admissible abstraction heuristic for cost-optimal planning

1. Start with the projection over variables: \( v_1, v_2, v_3, v_4 \)
2. Merge: replace \( \Theta_i \) and \( \Theta_j \) by their product
3. Shrink: replace \( \Theta_i \) by its abstraction \( \alpha(\Theta_i) \)

\[ \vdots \]

\[ \vdots \]

\[ \vdots \]

\[ \vdots \]
Merge-and-Shrink Heuristic

An admissible abstraction heuristic for cost-optimal planning

1. Start with the projection over variables: $v_1, v_2, v_3, v_4$
2. Merge: replace $\Theta_i$ and $\Theta_j$ by their product
3. Shrink: replace $\Theta_i$ by its abstraction $\alpha(\Theta_i)$

\[ \mathcal{L} \]

\[ \alpha_{1,2} \] \hspace{1cm} \[ \alpha_{3,4} \]
Merge-and-Shrink Heuristic

An admissible abstraction heuristic for cost-optimal planning

1. Start with the projection over variables: $v_1, v_2, v_3, v_4$
2. Merge: replace $\Theta_i$ and $\Theta_j$ by their product
3. Shrink: replace $\Theta_i$ by its abstraction $\alpha(\Theta_i)$

\[
\mathcal{L}
\]

$\alpha_{1,2} \otimes \alpha_{3,4}$
Simulation-Based Dominance Pruning

Label-dominance simulation (Torralba and Hoffmann, IJCAI 2015):

1. Use M&S to compute a partition of the problem: \( \{ \Theta_1, \ldots, \Theta_k \} \)
2. Compute label-dominance simulation relation: \( \{ \preceq_1, \ldots, \preceq_k \} \)
   - **Label dominance**: \( l \) dominates \( l' \) in \( \Theta_i \) if for any \( s \rightarrow t \) exists \( s \rightarrow t' \) s.t. \( t \preceq t' \)
   - **State dominance** \( s \preceq t \): For any \( s \rightarrow s' \), exists \( t \rightarrow t' \) s.t.:
     - \( t \preceq t' \)
     - \( c(l') \leq c(l) \)
     - \( l' \) dominates \( l \) in the rest of the problem
3. In \( A^* \), prune any \( s \) s.t. \( s \preceq t, g(s) \geq g(t) \) for some \( t \)
Merge-and-Shrink Framework (Sievers et al. 2014)

Θ₁  Θ₂  Θ₃  Θ₄  Global Θ
Merge-and-Shrink Framework (Sievers et al. 2014)

FDR task: $\langle V, O, I, G \rangle$

$v_1$  $v_2$  $v_3$  $v_4$

State space
Merge-and-Shrink Framework (Sievers et al. 2014)

M&S: Framework for transformation of planning tasks

<table>
<thead>
<tr>
<th>Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Merge</td>
</tr>
<tr>
<td>Shrink</td>
</tr>
<tr>
<td>Exact Label Reduction</td>
</tr>
<tr>
<td>Bisimulation shrinking</td>
</tr>
<tr>
<td>Reachability pruning</td>
</tr>
</tbody>
</table>

Global $\Theta$
**Merge-and-Shrink Framework (Sievers et al. 2014)**

\[ \Theta_1 \quad \Theta_2 \quad \Theta_3 \quad \Theta_4 \quad \text{Global } \Theta \]

**M&S: Framework for transformation of planning tasks**

<table>
<thead>
<tr>
<th>Operation</th>
<th>Transformation to global LTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Merge</td>
<td>None</td>
</tr>
<tr>
<td>Shrink</td>
<td>Abstraction</td>
</tr>
<tr>
<td>Exact Label Reduction</td>
<td>Change labels</td>
</tr>
<tr>
<td>Bisimulation shrinking</td>
<td>Preserves $h^*$</td>
</tr>
<tr>
<td>Reachability pruning</td>
<td>Keeps reachable/solvable part</td>
</tr>
</tbody>
</table>
Merge-and-Shrink Framework (Sievers et al. 2014)

M&S: Framework for transformation of planning tasks

<table>
<thead>
<tr>
<th>Operation</th>
<th>Transformation to global LTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Merge</td>
<td>None</td>
</tr>
<tr>
<td>Shrink</td>
<td>Abstraction</td>
</tr>
<tr>
<td>Exact Label Reduction</td>
<td>Change labels</td>
</tr>
<tr>
<td>Bisimulation shrinking</td>
<td>Preserves $h^*$</td>
</tr>
<tr>
<td>Reachability pruning</td>
<td>Keeps reachable/solvable part</td>
</tr>
<tr>
<td>Subsumed transition pruning</td>
<td>Preserves $h^*$</td>
</tr>
</tbody>
</table>
Plan Preserving Transformations of Planning Tasks

Plan preserving:
1. Does not add any new optimal plan to the task
2. At least one optimal plan for the original task is preserved \( (h^*(I)) \)

Unreachable/dead-end pruning is plan preserving

In this paper: subsumed transition pruning
- remove transitions from M&S transition systems
- globally \( h \)-preserving \( (h^*(s) \text{ for every } s) \)
Subsumed Transition Pruning

Definition (Subsumed transition)

\( s_i \xrightarrow{l} t_i \) is \textbf{subsumed} by \( s_i \xrightarrow{l'} t'_i \) if:

1. \( t_i \preceq t'_i \) and
2. \( c(l') \leq c(l) \) and
3. \( l' \) dominates \( l \) in all \( \Theta_j \) for \( j \neq i \).

Thm: Remove subsumed transitions is globally \( h \)-preserving.
Subsumed Transition Pruning

Definition (Subsumed transition)

$s_i \xrightarrow{l} t_i$ is subsumed by $s_i \xrightarrow{l'} t'_i$ if:

1. $t_i \preceq t'_i$ and
2. $c(l') \leq c(l)$ and
3. $l'$ dominates $l$ in all $\Theta_j$ for $j \neq i$.

Thm: Remove subsumed transitions is globally $h$-preserving
Subsumed Transition Pruning

**Definition (Subsumed transition)**

\[ s_i \overset{l}{\rightarrow} t_i \text{ is subsumed by } s_i \overset{l'}{\rightarrow} t'_i \text{ if:} \]

1. \( t_i \preceq t'_i \) and
2. \( c(l') \leq c(l) \) and
3. \( l' \) dominates \( l \) in all \( \Theta_j \) for \( j \neq i \).

Thm: Remove subsumed transitions is globally \( h \)-preserving

\[ \begin{align*}
A \rightarrow B \rightarrow C \rightarrow D \\
\overline{\mathcal{I}} \rightarrow E \rightarrow \mathcal{G}
\end{align*} \]

\[ \mathcal{I} \rightarrow A \text{ is subsumed by } \mathcal{I} \rightarrow E \\
\mathcal{G} \rightarrow D \text{ is subsumed by } \mathcal{G} \rightarrow E \]
**Subsumed Transition Pruning**

**Definition (Subsumed transition)**

\( s_i \xrightarrow{I} t_i \) is **subsumed** by \( s_i \xrightarrow{I'} t'_i \) if:

1. \( t_i \preceq t'_i \) and
2. \( c(l') \leq c(l) \) and
3. \( l' \) dominates \( l \) in all \( \Theta_j \) for \( j \neq i \).

**Thm:** Remove subsumed transitions is globally \( h \)-preserving

\[ \begin{align*}
    &A \quad B \quad C \quad D \\
    &\downarrow \quad \downarrow \quad \downarrow \quad \downarrow \\
    &I \quad E \quad G
\end{align*} \]

\( I \rightarrow A \) is subsumed by \( I \rightarrow E \)
\( G \rightarrow D \) is subsumed by \( G \rightarrow E \)
A, B, C, D become unreachable
Example: Subsumed Transition Pruning

$\Theta_1$: $s_1 \xrightarrow{l} t_1 \quad (s_1, s_2) \xrightarrow{l} (s_1, t_2)$

$\Theta_2$: $s_2 \xrightarrow{l} t_2 \quad (t_1, s_2) \xrightarrow{l} (t_1, t_2)$

$\Theta_1 \otimes \Theta_2$: $s_1 \xrightarrow{l} t_1 \quad (s_1, s_2) \xrightarrow{l} (s_1, t_2)$

Don't remove a transition if the label dominance changes!
Example: Subsumed Transition Pruning

$\Theta_1$:  
$s_1 \xrightarrow{l} t_1$

$\Theta_2$:  
$s_2 \xrightarrow{l''} t_2$

$\Theta_1 \otimes \Theta_2$:  
$(s_1, s_2) \xrightarrow{l} (s_1, t_2)$

$(t_1, s_2) \xrightarrow{l''} (t_1, t_2)$

$s_1 \xrightarrow{l} t_1$ is subsumed by $s_1 \xrightarrow{l''} t_1$
Example: Subsumed Transition Pruning

\[ \Theta_1: \]
\[ s_1 \xrightarrow{l} t_1 \]
\[ s_1 \xrightarrow{l'} t_1' \]

\[ \Theta_2: \]
\[ s_2 \xrightarrow{l} t_2 \]
\[ s_2 \xrightarrow{l'} t_2' \]

\[ \Theta_1 \otimes \Theta_2: \]
\[ (s_1, s_2) \xrightarrow{l} (s_1, t_2) \]
\[ (s_1, s_2) \xrightarrow{l'} (s_1, t_2') \]

- \( s_1 \xrightarrow{l} t_1 \) is subsumed by \( s_1 \xrightarrow{l'} t_1 \)
- \( s_2 \xrightarrow{l'} t_2 \) is subsumed by \( s_2 \xrightarrow{l} t_2 \)

Don't remove a transition if the label dominance changes!
Example: Subsumed Transition Pruning

\[ \Theta_1: \]
\[ s_1 \xrightarrow{l} t_1 \]

\[ \Theta_2: \]
\[ s_2 \xrightarrow{l'} t_2 \]

\[ \Theta_1 \otimes \Theta_2: \]
\[ (s_1, s_2) \xrightarrow{l} (s_1, t_2) \]
\[ (t_1, s_2) \xrightarrow{l'} (t_1, t_2) \]

- \( s_1 \xrightarrow{l} t_1 \) is subsumed by \( s_1 \xrightarrow{l'} t_1 \)
- \( s_2 \xrightarrow{l'} t_2 \) is subsumed by \( s_2 \xrightarrow{l} t_2 \)

Don’t remove a transition if the label dominance changes!
Taking Advantage of Plan-Preserving Transformations

1. Search task $\Pi'$ instead of $\Pi$
   - implementation overhead (future work)

2. Remove dead operators: after subsumed transition and unreachability pruning

3. M&S heuristics: If $\Theta'$ is a plan-preserving transformation of $\Theta$,
   - abstractions of $\Theta'$ are not admissible for $\Theta$
   - It's not a bug, it's a feature!!!

   $\Rightarrow$ less expanded states
   - globally admissible (preserve $h^*$ in at least one optimal plan)
   - $A^*$ returns optimal solutions

Subsumed transition pruning + unreachability analysis must be applied before any shrinking (except bisimulation)

Torralba, Kissmann

From Dominance to Irrelevance Pruning

SoCS 2015
Taking Advantage of Plan-Preserving Transformations

1. Search task $\Pi'$ instead of $\Pi$
   - implementation overhead (future work)

2. Remove dead operators:
   - after subsumed transition and unreachability pruning

It's not a bug, it's a feature!!!
→ less expanded states
globally admissible
$A^*$ returns optimal solutions

Subsumed transition pruning + unreachability analysis must be applied before any shrinking (except bisimulation)
Taking Advantage of Plan-Preserving Transformations

1. Search task $\Pi'$ instead of $\Pi$
   - implementation overhead (future work)

2. Remove dead operators:
   - after subsumed transition and unreachability pruning

3. M&S heuristics: If $\Theta'$ is a plan-preserving transformation of $\Theta$, abstractions of $\Theta'$ are not admissible for $\Theta$
Taking Advantage of Plan-Preserving Transformations

1. Search task $\Pi'$ instead of $\Pi$
   - implementation overhead (future work)

2. Remove dead operators:
   - after subsumed transition and unreachability pruning

3. M&S heuristics: If $\Theta'$ is a plan-preserving transformation of $\Theta$, abstractions of $\Theta'$ are not admissible for $\Theta$
   - It’s not a bug, it’s a feature!!! $\rightarrow$ less expanded states
   - **globally admissible** (preserve $h^*$ in at least one optimal plan) $\Rightarrow$ $A^*$ returns optimal solutions

Subsumed transition pruning + unreachability analysis **must be applied before any shrinking** (except bisimulation)
Similarity Shrinking

- Shrink $s$, $t$ iff $s \preceq t$ and $t \preceq s$
- Globally $h$-preserving $\Rightarrow$ derives perfect heuristics
- Coarser than bisimulation ($s$ and $s'$ are similar but not bisimilar)

Redundant with subsumed transition pruning (mod label reduction)
Experiments

- **Configuration \( P^i \):**
  - Incremental computation: recompute simulation after each merge
  - No label reduction, no shrinking

- Preprocess successful in 1463 of 1612 tasks
- Takes around 100s but up to 500-1000s in larger tasks
  - Suitable for optimal but not for satisficing planning
Experiments: M&S Heuristic

Expanded nodes

Total time (s)

M&S with $P^e$

Torralba, Kissmann

From Dominance to Irrelevance Pruning

SoCS 2015
### Experiments: Removing Irrelevant Operators

<table>
<thead>
<tr>
<th>Domain</th>
<th>% pruned operators</th>
<th>Coverage LM-cut</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$P^i$ $h^2$ $h^2 + P^i$</td>
<td>- $P^i$ $h^2$ $h^2 + P^i$</td>
</tr>
<tr>
<td>Floortile11</td>
<td>28 38 38</td>
<td>7 1 7 7</td>
</tr>
<tr>
<td>Logistics00</td>
<td>67 0 67</td>
<td>20 1 0 1</td>
</tr>
<tr>
<td>NoMystery</td>
<td>49 23 49</td>
<td>14 4 0 4</td>
</tr>
<tr>
<td>ParcPrint11</td>
<td>77 70 79</td>
<td>13 6 4 6</td>
</tr>
<tr>
<td>Rovers</td>
<td>71 0 71</td>
<td>7 3 0 2</td>
</tr>
<tr>
<td>Satellite</td>
<td>50 0 50</td>
<td>7 2 0 2</td>
</tr>
<tr>
<td>TPP</td>
<td>25 56 61</td>
<td>6 1 0 1</td>
</tr>
<tr>
<td>Trucks</td>
<td>90 38 90</td>
<td>10 1 0 1</td>
</tr>
<tr>
<td>Woodwk11</td>
<td>89 51 88</td>
<td>12 8 3 8</td>
</tr>
<tr>
<td>Total (1612)</td>
<td>32 23 42</td>
<td>833 29 46 65</td>
</tr>
</tbody>
</table>

+13 problems for symbolic bidirectional uniform-cost search (over 964)
Experiments: Comparison with State of the Art

HHJ (Haslum, Helmert, and Jonsson ICAPS 2013)

- Analyzes path subsumption in DTGs
- Current implementation only applicable to unary domains

<table>
<thead>
<tr>
<th>Domain</th>
<th>Operators $P^i$</th>
<th>LM-Cut $P^i$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Blocksworld</td>
<td>0.01 0.81</td>
<td>28 28 35</td>
</tr>
<tr>
<td>Driverlog</td>
<td>0.05 0.05</td>
<td>13 13 14</td>
</tr>
<tr>
<td>Logistics00</td>
<td>0.65 0.52</td>
<td>20 21 21</td>
</tr>
<tr>
<td>Logistics98</td>
<td>0.38 0.09</td>
<td>6 6 6</td>
</tr>
<tr>
<td>Miconic</td>
<td>0.58 0.57</td>
<td>141 142 142</td>
</tr>
<tr>
<td>Total</td>
<td>208 210 218</td>
<td></td>
</tr>
</tbody>
</table>
Conclusions

Take home messages:

1. **M&S** is suitable for transformation of planning tasks
2. Simulation relations useful for:
   - Subsumed transition pruning $\rightarrow$ very good in practice!
   - Similarity shrinking:
     - perfect shrinking better than bisimulation but...
     - redundant with subsumed transition pruning + bisimulation
3. **Irrelevance pruning** greatly simplifies planning tasks

Future work:

- Extensions of label-dominance simulation
- Path subsumption
- More types of problem transformations
Thanks for your attention!

Questions?