Subset-Saturated Transition Cost Partitioning

Dominik Drexler\textsuperscript{1,2}
Jendrik Seipp\textsuperscript{1,3}
David Speck\textsuperscript{2}

\textsuperscript{1}Linköping University, \textsuperscript{2}University of Freiburg, \textsuperscript{3}University of Basel
In a Nutshell

- Optimal classical planning
- A* search with admissible heuristic
- Cost partitioning [Katz and Domshlak, 2008]
- Saturated cost partitioning [Seipp et al., 2020]
- Main contribution: unify two orthogonal generalizations [Keller et al., 2016, Seipp and Helmert, 2019]

<table>
<thead>
<tr>
<th>generalization (1)</th>
<th>all states</th>
<th>subset of states</th>
</tr>
</thead>
<tbody>
<tr>
<td>operators</td>
<td>saturated operator CP</td>
<td>subset-saturated operator CP</td>
</tr>
<tr>
<td>transitions</td>
<td>saturated transition CP</td>
<td><strong>subset-saturated transition CP</strong></td>
</tr>
</tbody>
</table>
Cost Partitioning

- How to **admissibly** combine the information of collection of admissible heuristics $h_1, \ldots, h_n$?

- Consider operator cost function $ocf : L \rightarrow \mathbb{R}^+$ of planning task

- Heuristic admissible if $h(ocf, s) \leq h^*(ocf, s)$ for all states $s$

- $\max(h_1(ocf, s), \ldots, h_n(ocf, s))$ ✔

- $h_1(ocf, s) + \ldots + h_n(ocf, s)$ ✗

- **Cost partitioning**: $h_1(ocf_1, s) + \ldots + h_n(ocf_n, s)$ ✔
  
  if $\sum_{i=1}^{n} oc f_i(l) \leq oc f(l)$ for all $l \in L$

How to find cost partitioning that yields strong heuristic?
Saturated Cost Partitioning

\[
\text{for heuristic } h \text{ in sequence } h_1, \ldots, h_n \text{ do}
\]
\[
ocf_i \leftarrow \text{saturate}(h, \ocf)
\]
\[
ocf \leftarrow \ocf - \ocf_i
\]
\end{for}

Example

\[
h_1(ocf_1, s_0) + h_2(ocf_2, s_0) = 1 + 1 = 2 \checkmark
\]
Generalizations of Saturated Cost Partitioning

<table>
<thead>
<tr>
<th>Generalization (1)</th>
<th>Operators</th>
<th>Transitions</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>saturated operator CP</td>
<td>saturated transition CP</td>
</tr>
<tr>
<td></td>
<td>subset-saturated operator CP</td>
<td>subset-saturated transition CP</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Generalization (2)</th>
<th>all states</th>
<th>subset of states</th>
</tr>
</thead>
<tbody>
<tr>
<td>operators</td>
<td>saturated operator CP</td>
<td>subset-saturated operator CP</td>
</tr>
<tr>
<td>transitions</td>
<td>saturated transition CP</td>
<td><strong>subset-saturated transition CP</strong></td>
</tr>
</tbody>
</table>

(1) Costs partitioned among transitions [Keller et al., 2016]
- Function \( \text{saturate} \) returns \( tcf \_i : T \rightarrow \mathbb{R} \) instead of \( ocf \_i \);
- Intractable in the worst-case, often manageable

(2) Saturate for subset of states \( S' \subseteq S \) [Seipp and Helmert, 2019]
- E.g., reachable, single state
Further Contributions

1. Faster computation of $h(tcf, s)$
   - Reduce computations of abstract transition costs

2. Restrictions on $tcf_i$
   - Tractability depends on $tcf_i$
   - Focus computational effort on the subset of states $S'$

3. Future work: when to restrict to $ocf_i$?
## Experiments

<table>
<thead>
<tr>
<th></th>
<th>(1)</th>
<th>(2)</th>
<th>(3)</th>
<th>(4)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1) saturated operator CP</td>
<td>–</td>
<td>47</td>
<td>164</td>
<td>59</td>
</tr>
<tr>
<td>(2) subset-saturated operator CP</td>
<td>488</td>
<td>–</td>
<td>390</td>
<td>55</td>
</tr>
<tr>
<td>(3) saturated transition CP</td>
<td>345</td>
<td>236</td>
<td>–</td>
<td>34</td>
</tr>
<tr>
<td>(4) subset-saturated transition CP</td>
<td>683</td>
<td>400</td>
<td>482</td>
<td>–</td>
</tr>
<tr>
<td>Coverage (1827 total tasks)</td>
<td>1056</td>
<td>1061</td>
<td>1024</td>
<td><strong>1083</strong></td>
</tr>
</tbody>
</table>
Conclusions and Future Work

- Generalized saturated cost partitioning further
- Our planner is competitive with state of the art
- Future work: switching between the cost function type (using reasoning or learning techniques)
Optimal additive composition of abstraction-based admissible heuristics.

State-dependent cost partitionings for Cartesian abstractions in classical planning.

Subset-saturated cost partitioning for optimal classical planning.

Saturated cost partitioning for optimal classical planning.