Introduction to Cost Partitioning

Florian Pommerening

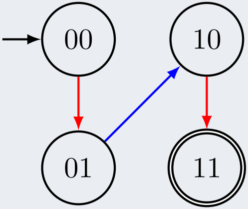
University of Basel, Switzerland

October 4, 2018

- One pattern database is good, so more must be better!?
- Yes, but how can we combine them?
  - maximum
  - ~~sum~~
  - ?

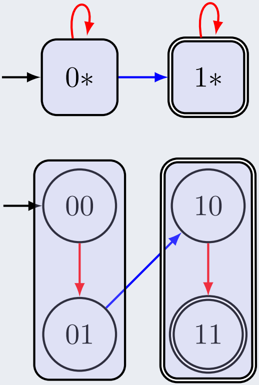
PDBs in Planning

Projections of Planning Tasks



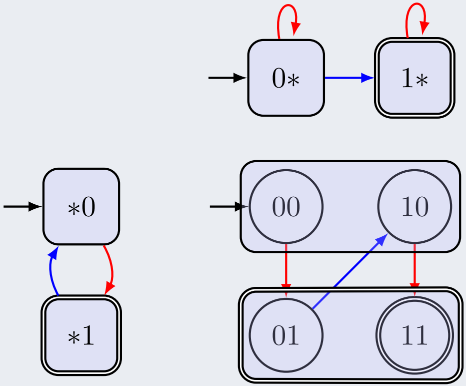
PDBs in Planning

Projections of Planning Tasks



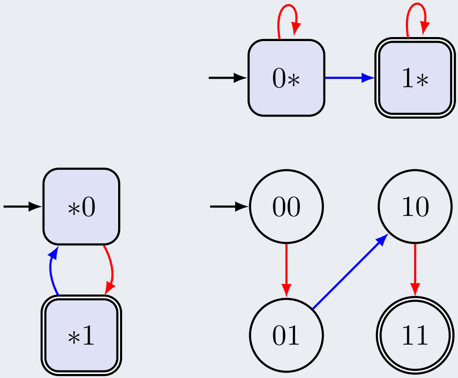
PDBs in Planning

Projections of Planning Tasks



PDBs in Planning

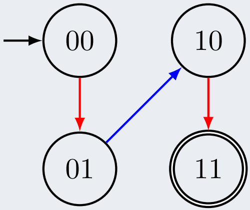
Projections of Planning Tasks



Cost Partitioning Beyond Planning

- more general setting
  - solution is sequence of steps
  - each step has a cost
  - solution cost = sum of step costs
  - admissible heuristics
- example:
  multiple sequence alignment

Planning Task



History of Cost Partitioning

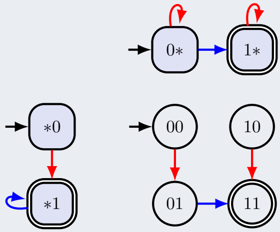
Disjoint Pattern Databases

Disjoint pattern databases

[Korf & Felner, 2002; Edelkamp, 2001]

- no operator influences more than one pattern
- PDBs are additive
- proof idea:
  - consider an optimal plan
  - each PDB underestimates contribution of a set of operators
  - sets do not overlap

Example



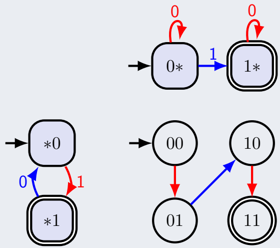
Operator Partitioning

Operator Partitioning

[Haslum, Bonet & Geffner, 2005; Edelkamp, 2006]

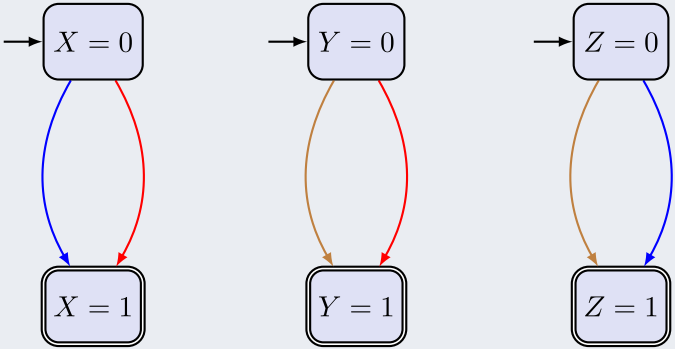
- partition operators into sets
  - one set for each PDB
  - assigning each operator to a PDB
- ignore cost of all other operators
- PDBs are additive
- same proof idea

Example



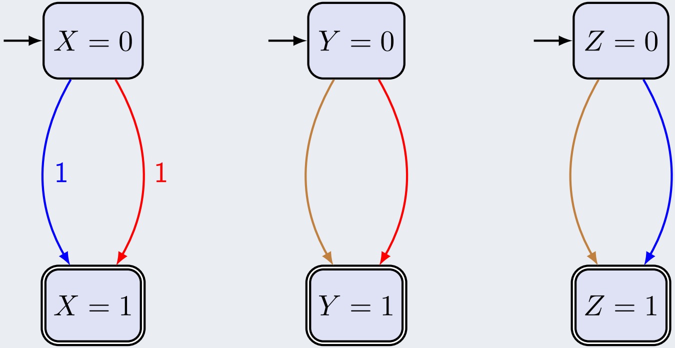
Bad Case for Operator Partitioning

Example [Bonet and Helmert, 2010]



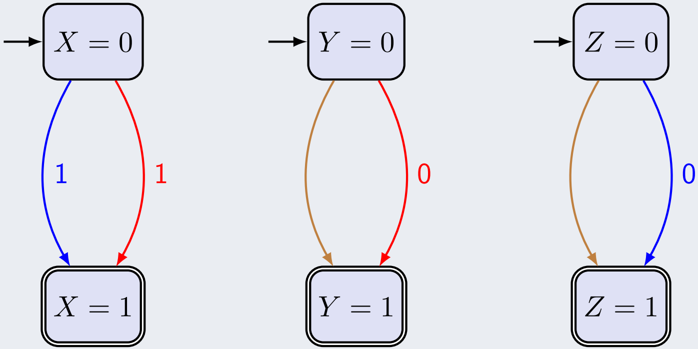
Bad Case for Operator Partitioning

Example [Bonet and Helmert, 2010]



Bad Case for Operator Partitioning

Example [Bonet and Helmert, 2010]



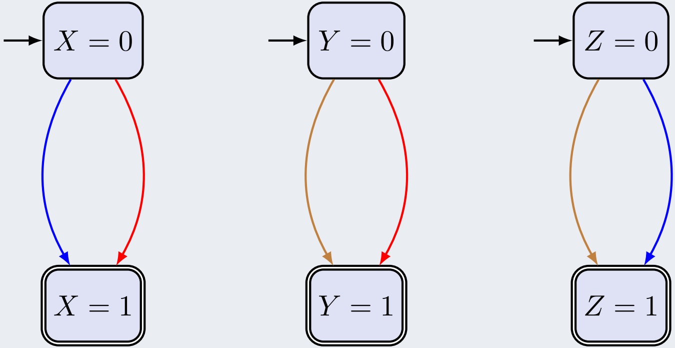
Cost Partitioning

Main Idea of Cost Partitioning

- operator partitioning
  - full cost in one PDB
  - zero cost in all other PDBs
- cost partitioning
  - fraction of the cost in each PDB

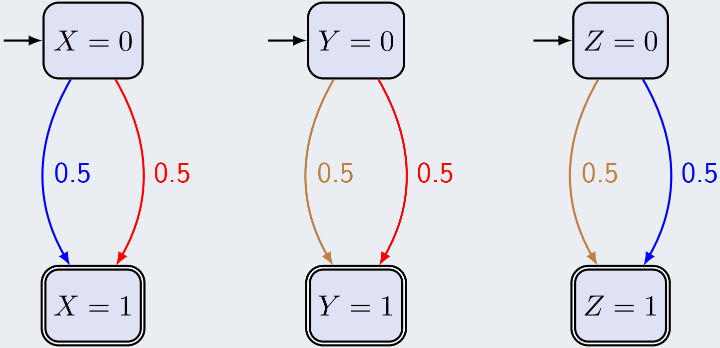
Example

Combining projections with cost partitioning



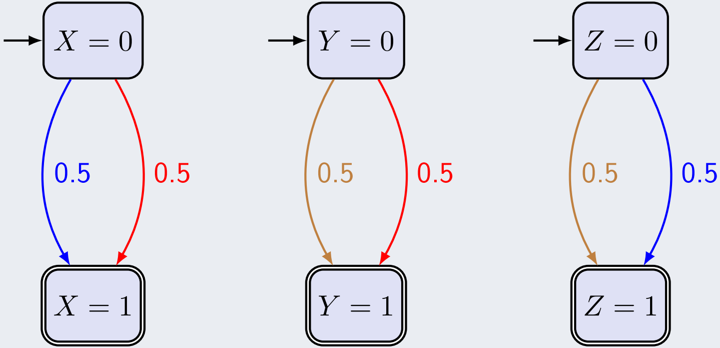
Example

Combining projections with cost partitioning



Example

Combining projections with cost partitioning



Heuristic value: $0.5 + 0.5 + 0.5 = 1.5$ (can be rounded up to 2)

Definition

Cost Partitioning

- compute each heuristic under its own cost function

$$c_i : \mathcal{O} \to \mathbb{R}_0^+$$

- require for all operators o

$$c_1(o) + \cdots + c_n(o) \leq c(o)$$

⇝ sum of resulting estimates is admissible

Admissibility

Proof of Admissibility

$$\sum_i h_i(c_i)$$

Admissibility

Proof of Admissibility

$$\sum_i h_i(c_i) \leq \sum_i h^*(c_i) \qquad \text{(h_i is admissible)}$$

Admissibility

Proof of Admissibility

$$\sum_i h_i(c_i) \leq \sum_i h^*(c_i) \qquad (h_i \text{ is admissible})$$

$$\leq \sum_i c_i(\pi^*) \qquad (\pi^* \text{ is optimal under } c)$$

Admissibility

Proof of Admissibility

$$\begin{aligned} \sum_i h_i(c_i) &\leq \sum_i h^*(c_i) && (h_i \text{ is admissible}) \\ &\leq \sum_i c_i(\pi^*) && (\pi^* \text{ is optimal under } c) \\ &= \sum_i \sum_j c_i(\pi_j^*) && (\text{definition}) \end{aligned}$$

Admissibility

Proof of Admissibility

$$
\begin{aligned}
\sum_i h_i(c_i) &\leq \sum_i h^*(c_i) && \text{(h_i is admissible)} \\
&\leq \sum_i c_i(\pi^*) && \text{(π^* is optimal under c)} \\
&= \sum_i \sum_j c_i(\pi_j^*) && \text{(definition)} \\
&= \sum_j \sum_i c_i(\pi_j^*) && \text{(arithmetic)}
\end{aligned}
$$

Admissibility

Proof of Admissibility

$$\begin{aligned}
\sum_i h_i(c_i) &\leq \sum_i h^*(c_i) && \text{(h_i is admissible)} \\
&\leq \sum_i c_i(\pi^*) && \text{(π^* is optimal under c)} \\
&= \sum_i \sum_j c_i(\pi_j^*) && \text{(definition)} \\
&= \sum_j \sum_i c_i(\pi_j^*) && \text{(arithmetic)} \\
&\leq \sum_j c(\pi_j^*) && \text{(cost partitioning)}
\end{aligned}$$

Admissibility

Proof of Admissibility

$$
\begin{aligned}
\sum_i h_i(c_i) &\leq \sum_i h^*(c_i) && (h_i \text{ is admissible})\\
&\leq \sum_i c_i(\pi^*) && (\pi^* \text{ is optimal under } c)\\
&= \sum_i \sum_j c_i(\pi_j^*) && (\text{definition})\\
&= \sum_j \sum_i c_i(\pi_j^*) && (\text{arithmetic})\\
&\leq \sum_j c(\pi_j^*) && (\text{cost partitioning})\\
&= c(\pi^*) = h^*(c) && (\text{definition})
\end{aligned}
$$

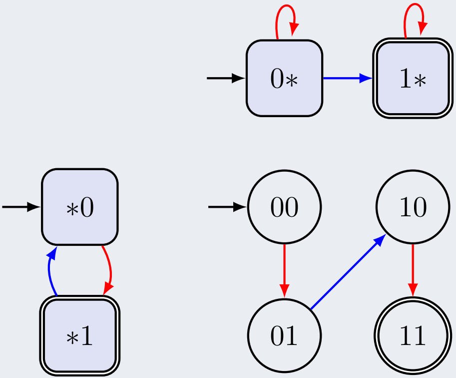
General Cost Partitioning

general cost partitioning [Pommerening et al., 2015]

- so far: all cost functions non-negative
- can also allow negative costs
- still admissible (same proof)
- small change, big impact

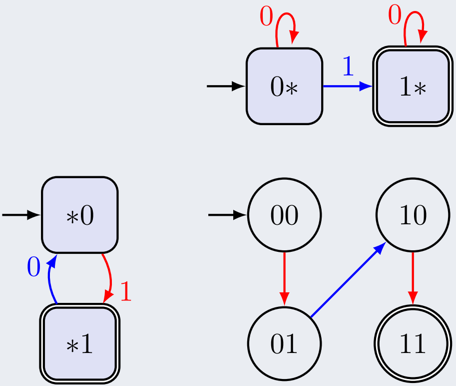
Example

Negative Costs can increase the heuristic value



Example

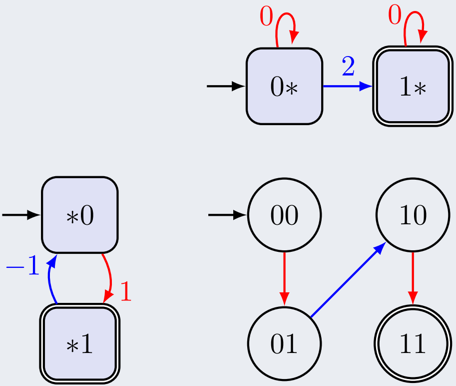
Negative Costs can increase the heuristic value



Heuristic value: $1 + 1 = 2$

Example

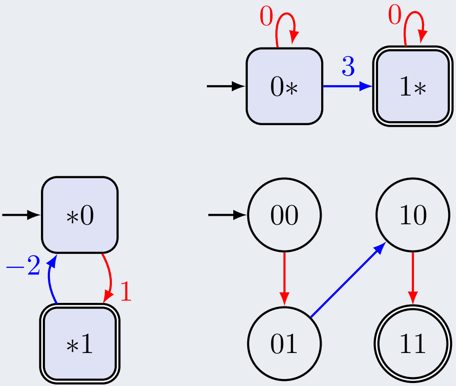
Negative Costs can increase the heuristic value



Heuristic value: $1 + 2 = 3$

Example

Negative Costs can increase the heuristic value



Heuristic value: $-\infty + 3 = -\infty$

Take Home Messages

Take Home Messages

Cost Partitioning

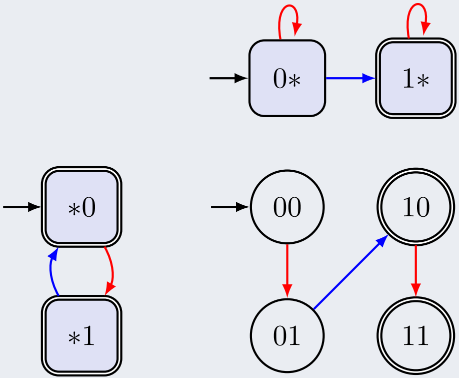
- combines admissible heuristics
- partitions cost function: $\sum_i c_i \leq c$
- results in admissible estimate

But how can you find the right cost functions c_i?
$\rightsquigarrow$ Jendrik's talk

Additional Slides

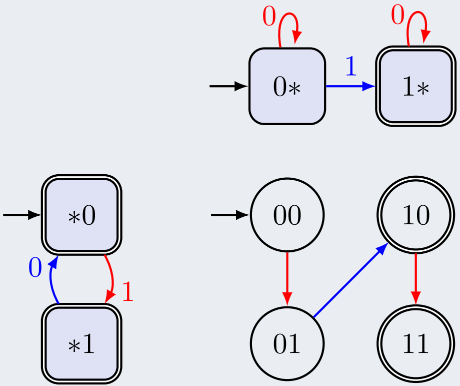
Example

Projections to non-goal variables can contribute



Example

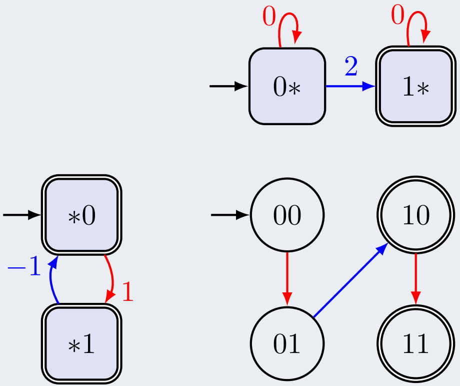
Projections to non-goal variables can contribute



Heuristic value: $0 + 1 = 1$

Example

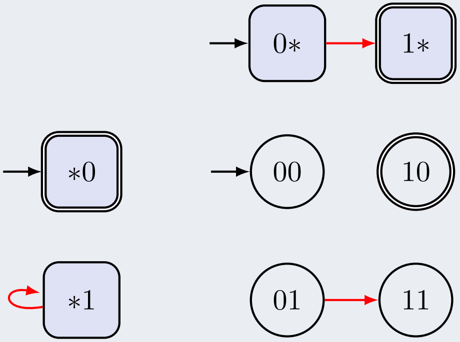
Projections to non-goal variables can contribute



Heuristic value: $0 + 2 = 2$

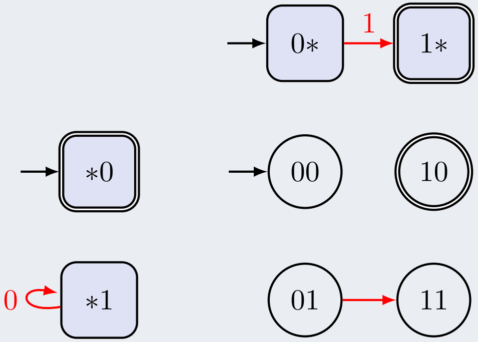
Example 3

Unsolvability detected even if all abstractions are solvable



Example 3

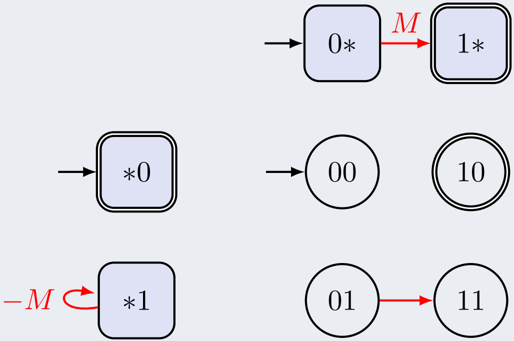
Unsolvability detected even if all abstractions are solvable



Heuristic value: $0 + 1 = 1$

Example 3

Unsolvability detected even if all abstractions are solvable



Heuristic value: $0 + M = M$