

Planning and Optimization

F8. Optimal and General Cost-Partitioning

Malte Helmert and Gabriele Röger

Universität Basel

December 9, 2024

Planning and Optimization

December 9, 2024 — F8. Optimal and General Cost-Partitioning

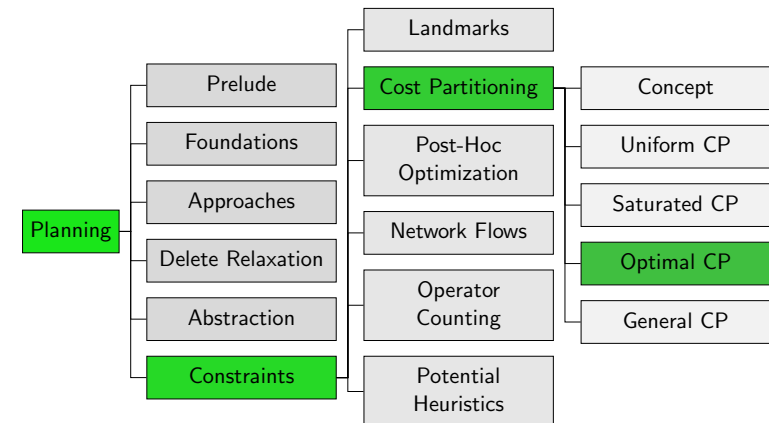
F8.1 Optimal Cost Partitioning

F8.2 General Cost Partitioning

F8.3 Summary

F8.1 Optimal Cost Partitioning

Content of the Course



Optimal Cost Partitioning: General Approach

- ▶ Can we find a better cost partitioning than with the uniform or saturation strategy? Even an **optimal** one?
- ▶ Idea: exploit linear programming
 - ▶ Use variables for cost of each operator in each task copy
 - ▶ Express heuristic values with linear constraints
 - ▶ Maximize sum of heuristic values subject to these constraints

LPs known for

- ▶ abstraction heuristics (not covered in this course)
- ▶ disjunctive action landmarks (now)

Optimal Cost Partitioning for Landmarks: Basic Version

- ▶ Use an LP that covers the heuristic computation and the cost partitioning.
- ▶ LP variable $C_{L,o}$ for cost of operator o in induced task for disjunctive action landmark L (cost partitioning)
- ▶ LP variable $Cost_L$ for cost of disjunctive action landmark L in induced task (value of individual heuristics)

Optimal Cost Partitioning for Landmarks: Basic LP

Variables

Non-negative variable $Cost_L$ for each disj. action landmark $L \in \mathcal{L}$

Non-negative variable $C_{L,o}$ for each $L \in \mathcal{L}$ and operator o

Objective

Maximize $\sum_{L \in \mathcal{L}} Cost_L$

Subject to

$$\sum_{L \in \mathcal{L}} C_{L,o} \leq cost(o) \quad \text{for all operators } o$$

$$Cost_L \leq C_{L,o} \quad \text{for all } L \in \mathcal{L} \text{ and } o \in L$$

Optimal Cost Partitioning for Landmarks: Improved

- ▶ **Observation:** Explicit variables for cost partitioning not necessary.
- ▶ Use implicitly $cost_L(o) = Cost_L$ for all $o \in L$ and 0 otherwise.

Optimal Cost Partitioning for Landmarks: Improved LP

Variables

Non-negative variable $Cost_L$ for each disj. action landmark $L \in \mathcal{L}$

Objective

Maximize $\sum_{L \in \mathcal{L}} Cost_L$

Subject to

$$\sum_{L \in \mathcal{L}: o \in L} Cost_L \leq cost(o) \quad \text{for all operators } o$$

Example (1)

Example

Let Π be a planning task with operators o_1, \dots, o_4 and $cost(o_1) = 3, cost(o_2) = 4, cost(o_3) = 5$ and $cost(o_4) = 0$.

Let the following be disjunctive action landmarks for Π :

$$\mathcal{L}_1 = \{o_4\}$$

$$\mathcal{L}_2 = \{o_1, o_2\}$$

$$\mathcal{L}_3 = \{o_1, o_3\}$$

$$\mathcal{L}_4 = \{o_2, o_3\}$$

Example (2)

Example

Maximize $Cost_{\mathcal{L}_1} + Cost_{\mathcal{L}_2} + Cost_{\mathcal{L}_3} + Cost_{\mathcal{L}_4}$ subject to

$$[o_1] \quad Cost_{\mathcal{L}_2} + Cost_{\mathcal{L}_3} \leq 3$$

$$[o_2] \quad Cost_{\mathcal{L}_2} + Cost_{\mathcal{L}_4} \leq 4$$

$$[o_3] \quad Cost_{\mathcal{L}_3} + Cost_{\mathcal{L}_4} \leq 5$$

$$[o_4] \quad Cost_{\mathcal{L}_1} \leq 0$$

$$Cost_{\mathcal{L}_i} \geq 0 \quad \text{for } i \in \{1, 2, 3, 4\}$$

Optimal Cost Partitioning for Landmarks (Dual view)

Variables

Non-negative variable $Applied_o$ for each operator o

Objective

Minimize $\sum_o Applied_o \cdot cost(o)$

Subject to

$$\sum_{o \in L} Applied_o \geq 1 \quad \text{for all landmarks } L$$

Minimize “plan cost” with all landmarks satisfied.

Example: Dual View

Example (Optimal Cost Partitioning: Dual View)

Minimize $3\text{Applied}_{o_1} + 4\text{Applied}_{o_2} + 5\text{Applied}_{o_3}$ subject to

$$\begin{aligned} \text{Applied}_{o_4} &\geq 1 \\ \text{Applied}_{o_1} + \text{Applied}_{o_2} &\geq 1 \\ \text{Applied}_{o_1} + \text{Applied}_{o_3} &\geq 1 \\ \text{Applied}_{o_2} + \text{Applied}_{o_3} &\geq 1 \\ \text{Applied}_{o_i} &\geq 0 \quad \text{for } i \in \{1, 2, 3, 4\} \end{aligned}$$

This is equal to the LP relaxation of the MHS heuristic

Reminder: LP Relaxation of MHS heuristic

Example (Minimum Hitting Set)

minimize $3X_{o_1} + 4X_{o_2} + 5X_{o_3}$ subject to

$$\begin{aligned} X_{o_4} &\geq 1 \\ X_{o_1} + X_{o_2} &\geq 1 \\ X_{o_1} + X_{o_3} &\geq 1 \\ X_{o_2} + X_{o_3} &\geq 1 \end{aligned}$$

$$X_{o_1} \geq 0, \quad X_{o_2} \geq 0, \quad X_{o_3} \geq 0, \quad X_{o_4} \geq 0$$

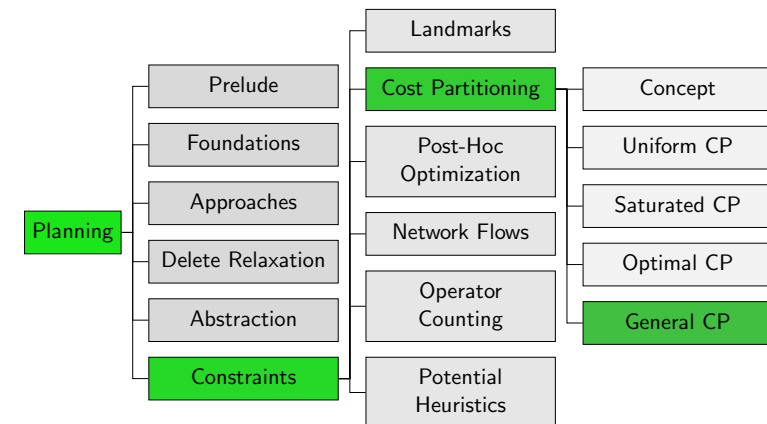
↪ optimal solution of LP relaxation:

$X_{o_4} = 1$ and $X_{o_1} = X_{o_2} = X_{o_3} = 0.5$ with objective value 6

↪ LP relaxation of MHS heuristic is **admissible**
and can be computed **polynomial time**

F8.2 General Cost Partitioning

Content of the Course



General Cost Partitioning

Cost functions are **usually non-negative**.

- ▶ We tacitly also required this for task copies
- ▶ Makes intuitively sense: original costs are non-negative
- ▶ But: not necessary for cost-partitioning!

General Cost Partitioning

Definition (General Cost Partitioning)

Let Π be a planning task with operators O .

A **general cost partitioning** for Π is a tuple $\langle cost_1, \dots, cost_n \rangle$, where

- ▶ $cost_i : O \rightarrow \mathbb{R}$ for $1 \leq i \leq n$ and
- ▶ $\sum_{i=1}^n cost_i(o) \leq cost(o)$ for all $o \in O$.

General Cost Partitioning: Admissibility

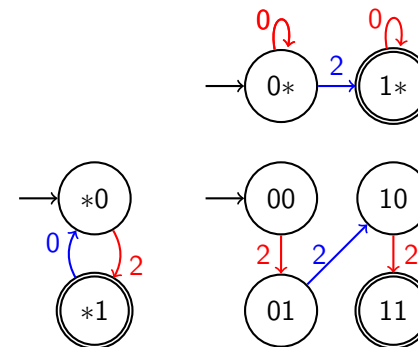
Theorem (Sum of Solution Costs is Admissible)

Let Π be a planning task, $\langle cost_1, \dots, cost_n \rangle$ be a **general cost partitioning** and $\langle \Pi_1, \dots, \Pi_n \rangle$ be the tuple of induced tasks.

Then the sum of the solution costs of the induced tasks is an **admissible heuristic** for Π , i.e., $\sum_{i=1}^n h_{\Pi_i}^* \leq h_{\Pi}^*$.

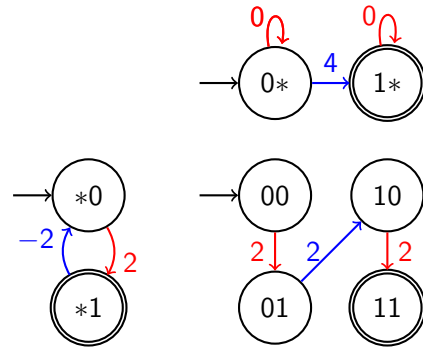
(Proof omitted.)

General Cost Partitioning: Example



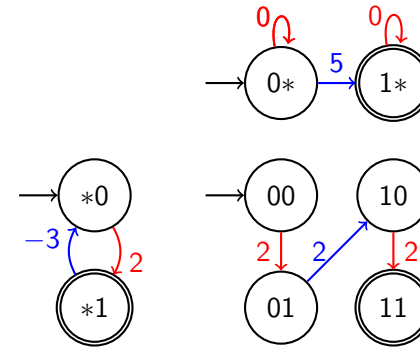
Heuristic value: $2 + 2 = 4$

General Cost Partitioning: Example



Heuristic value: $4 + 2 = 6$

General Cost Partitioning: Example



Heuristic value: $-\infty + 5 = -\infty$

F8.3 Summary

- ▶ For abstraction heuristics and disjunctive action landmarks, we know how to determine an **optimal cost partitioning**, using linear programming.
- ▶ Although solving a linear program is possible in polynomial time, the better heuristic guidance often does not outweigh the overhead (in particular for abstraction heuristics).
- ▶ In contrast to standard (non-negative) cost partitioning, **general cost partitioning** allows negative operators costs.
- ▶ General cost partitioning has the same relevant properties as non-negative cost partitioning but is more powerful.

Summary