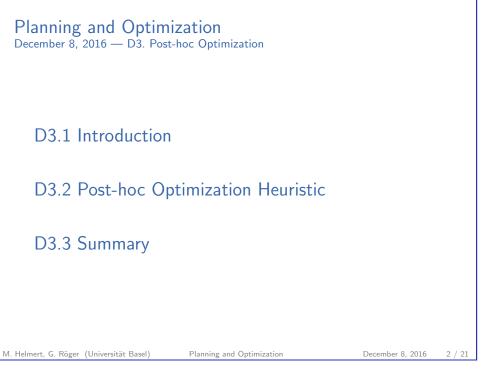
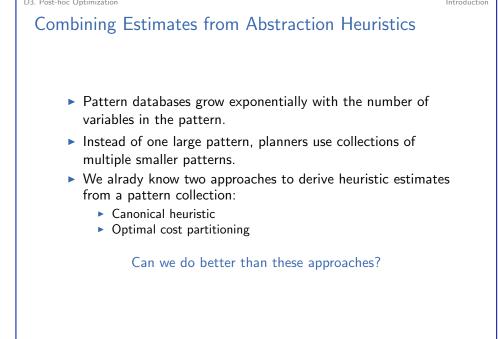


Planning and Optimization





M. Helmert, G. Röger (Universität Basel)

#### D3. Post-hoc Optimization

### Reminder: The Canonical Heuristic Function

If for a set of patterns no operator affects more than one pattern, the sum of the heuristic estimates is admissible.

### Definition (Canonical Heuristic Function)

Let  $\Pi$  be an FDR planning task. Let C be a pattern collection for  $\Pi$  and let cliques(C) denote the set of all maximal additive subsets of C. The canonical heuristic  $h^{C}$  for C is defined as

$$h^{\mathcal{C}}(s) = \max_{\mathcal{D} \in \textit{cliques}(\mathcal{C})} \sum_{P \in \mathcal{D}} h^{P}(s).$$

For a given pattern collection, the canonical heuristic is the best possible admissible heuristic not using cost partitioning.

Planning and Optimization

M. Helmert, G. Röger (Universität Basel)

December 8, 2016

Introduction

5 / 21

Example Task (1)

D3. Post-hoc Optimization

Example (Example Task) SAS<sup>+</sup> task  $\Pi = \langle V, I, O, \gamma \rangle$  with •  $V = \{A, B, C\}$  with dom $(v) = \{0, 1, 2, 3, 4\}$  for all  $v \in V$ •  $I = \{A \mapsto 0, B \mapsto 0, C \mapsto 0\}$ •  $O = \{inc_x^v \mid v \in V, x \in \{0, 1, 2\}\} \cup \{jump^v \mid v \in V\}$ •  $inc_x^v = \langle v = x, v := x + 1, 1 \rangle$ •  $jump^v = \langle \bigwedge_{v' \in V: v' \neq v} v' = 4, v := 3, 1 \rangle$ •  $\gamma = A = 3 \land B = 3 \land C = 3$ • Each optimal plan consists of three increment operators for each variable  $\rightsquigarrow h^*(I) = 9$ 

Planning and Optimization

#### D3. Post-hoc Optimization

# Reminder: Optimal Cost Partitioning for Abstractions

Optimal cost partitioning for abstractions...

- uses a state-specific LP to find the best possible cost partitioning, and sums up the heuristic estimates.
- ... dominates the canonical heuristic, i.e.. for the same pattern collection, it never gives lower estimates than h<sup>C</sup>.
- ... is very expensive to compute (recomputing the PDBs in every state).

M. Helmert, G. Röger (Universität Basel)

December 8, 2016

# D3. Post-hoc Optimization

# Example Task (2)

In projections on single variables we can reach the goal with a jump operator: h<sup>{A}</sup>(I) = h<sup>{B}</sup>(I) = h<sup>{C}</sup>(I) = 1.

Planning and Optimization

In projections on more variables, we need for each variable three applications of increment operators to reach the abstract goal from the abstract initial state: h<sup>{A,B}</sup>(I) = h<sup>{A,C}</sup>(I) = h<sup>{B,C}</sup>(I) = 6

#### Example (Canonical Heuristic)

$$C = \{\{A\}, \{B\}, \{C\}, \{A, B\}, \{A, C\}, \{B, C\}\}$$
$$h^{C}(s) = \max\{h^{\{A\}}(s) + h^{\{B\}}(s) + h^{\{C\}}(s), h^{\{A\}}(s) + h^{\{B,C\}}(s),$$
$$h^{\{B\}}(s) + h^{\{A,C\}}(s), h^{\{C\}}(s) + h^{\{A,B\}}(s)\}$$

Planning and Optimization

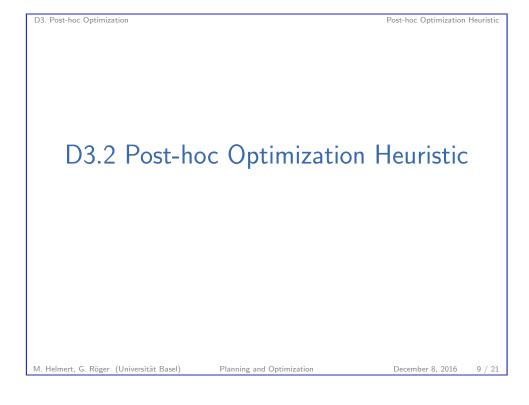
 $h^{\mathcal{C}}(I) = 7$ 

8 / 21

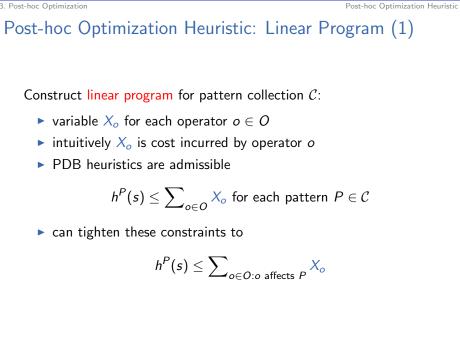
Introduction

6 / 21

Introductio



D3. Post-hoc Optimization



D3. Post-hoc C	Optimizatio
----------------	-------------

### Post-hoc Optimization Heuristic: Idea

Consider the example task:

- type-v operator: operator modifying variable v
- ▶  $h^{\{A,B\}} = 6$ 
  - $\Rightarrow$  any plan contains at least 6 operators of type A or B.
- ▶  $h^{\{A,C\}} = 6$ 
  - $\Rightarrow$  any plan contains at least 6 operators of type A or C.
- ▶  $h^{\{B,C\}} = 6$ 
  - $\Rightarrow$  any plan contains at least 6 operators of type B or C.

Planning and Optimization

 $\blacktriangleright$   $\Rightarrow$  at least 9 operators in any plan

Can we generalize this kind of reasoning?

M. Helmert, G. Röger (Universität Basel)

December 8, 2016

Post-hoc Optimization Heuristic

10 / 21

```
D3. Post-hoc Optimization
                                                                          Post-hoc Optimization Heuristic
  Post-hoc Optimization Heuristic: Linear Program (2)
      For pattern collection C:
      Variables
      X_o for each operator o \in O
      Objective
      Minimize \sum_{o \in O} X_o
      Subject to
               \sum_{o \in O:o \text{ affects } P} X_o \ge h^P(s) \quad \text{for all patterns } P \in \mathcal{C}
                                    X_o > 0 for all o \in O
M. Helmert, G. Röger (Universität Basel)
                                       Planning and Optimization
                                                                           December 8, 2016
                                                                                            12 / 21
```

December 8, 2016

11 / 21



# Post-hoc Optimization Heuristic: Simplifying the LP

- Reduce size of LP by aggregating variables which always occur together in constraints.
- Happens when several operators are relevant for exactly the same PDBs.

Planning and Optimization

- $\blacktriangleright$  Partitioning  $O\!/\!\!\sim$  induced by this equivalence relation
- One variable  $X_{[o]}$  for each  $[o] \in O/\!\!\sim$

M. Helmert, G. Röger (Universität Basel)

D3. Post-hoc Optimization

Post-hoc Optimization Heuristic

15 / 21

13 / 21

December 8, 2016

# Post-hoc Optimization Heuristic: Admissibility

### Theorem (Admissibility)

The post-hoc optimization heuristic is admissible.

#### Proof.

Let  $\Pi$  be a planning task and C be a pattern collection. Let  $\pi$  be an optimal plan for state *s* and let  $cost_{\pi}(O')$  be the cost incurred by operators from  $O' \subseteq O$  in  $\pi$ .

Setting each  $X_{[o]}$  to  $cost_{\pi}([o])$  is a feasible variable assignment: Constraints  $X_{[o]} \ge 0$  are satisfied. For each  $P \in C$ ,  $\pi$  is a solution in the abstract transition system and the sum in the corresponding constraint equals the cost of the "true" abstract state transitions (i.e.. not accounting for self-loops). As  $h^{P}(s)$  corresponds to the cost of an optimal solution in the abstraction, the inequality holds.

For this assignment the objective function has value  $h^*(s)$  (cost of  $\pi$ ), so the objective value of the LP is admissible.

### Post-hoc Optimization Heuristic: Definition

### Definition (Post-hoc Optimization Heuristic)

The post-hoc optimization heuristic  $h_C^{\text{PhO}}$  for pattern collection C is the objective value of the following linear program:

$$\begin{array}{ll} \text{Minimize} & \sum_{[o] \in \mathcal{O}/\!\!\sim} X_{[o]} \text{ subject to} \\ \\ & \sum_{[o] \in \mathcal{O}/\!\sim:o \text{ affects } P} X_{[o]} \geq h^P(s) & \text{for all } P \in \mathcal{C} \\ & X_{[o]} \geq 0 & \text{for all } [o] \in \mathcal{O}/\!\!\sim, \end{array}$$

where  $o \sim o'$  iff o and o' affect the same patterns in C.

- Precompute PDBs for all  $P \in C$ .
- ► Create LP for initial state.
- For each new state, just change the bounds  $h^{P}(s)$ .

Planning and Optimization

```
M. Helmert, G. Röger (Universität Basel)
```

December 8, 2016

14 / 21

16 / 21

```
D3. Post-hoc Optimization Heuristic: Insight

Corresponding dual program to h^{PhO} LP:

Maximize \sum_{P \in C} Y_P h^P(s) subject to

\sum_{P \in \mathcal{C}: o \text{ affects } P} Y_P \leq 1 for all [o] \in O/\sim

Y_P \geq 0 for all P \in C.

We compute a state-specific cost partitioning that can only scale

the operator costs within each heuristic by a factor Y_i.
```

M. Helmert, G. Röger (Universität Basel)

### Relation to Canonical Heuristic

#### Theorem

Consider the dual D of the LP solved by  $h_{C}^{PhO}$  in state s for a given pattern collection C. If we restrict the variables in D to integers. the objective value is the canonical heuristic value  $h^{\mathcal{C}}(s)$ .

#### Corollary

The post-hoc optimization heuristic  $h_c^{\text{PhO}}$  dominates the canonical heuristic  $h^{\mathcal{C}}$  for the same pattern collection  $\mathcal{C}$ .

Planning and Optimization

Planning and Optimization

M. Helmert, G. Röger (Universität Basel)

D3. Post-hoc Optimization

December 8, 2016

17 / 21

Summar

19 / 21

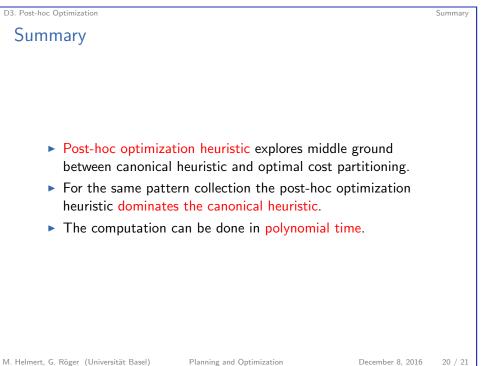
D3.3 Summary

## Post-hoc Optimization Heuristic: Remarks

- ▶ For the canonical heuristic, we need to find all maximal cliques, which is an NP-hard problem.
- ► The post-hoc optimization heuristic dominates the canonical heuristic and can be computed in polynomial time.
- ▶ With post-hoc optimization, we can handle much larger pattern collections than found with the iPDB procedure.
- For the approach it is better to use a large number of small patterns, e.g., all patterns up to size 2 that satisfy the same relevance criteria as used for the iPDB patterns.
- Post-hoc optimization is not limited to PDBs but there is a straightforward extension to any admissible heuristic for which we can determine the "relevant" operators.

```
M. Helmert, G. Röger (Universität Basel)
```

Planning and Optimization December 8, 2016



18 / 21

