Evaluation Of Post-Hoc Optimization Constraints Under Altered Cost Functions

Presentation of Master's Thesis

Andreas Thüring

Examiner: Dr. Gabriele Röger
Supervisor: Dr. Florian Pommerening

February 11, 2019

Universität Basel

Setting

- Classical Planning / Heuristic Search
- Heuristics based on linear programming
  - optimal cost-partitioning (Katz and Domshlak, 2010),
  - state-equation heuristic (Bonet, 2013),
  - landmark constraints (Zhu and Givan, 2003),
  - post-hoc optimization constraints (Pommerening et al., 2013)
- Operator-counting (Pommerening et al., 2014): a framework for heuristics based on linear programming

Operator-Counting (Pommerening et al., 2014)

Objective Function

$$\text{minimize} \sum_{o \in O} cost(o) \cdot Count_o \text{ subject to } C$$

- $Count_o$ is an *operator-counting variable* for every operator,
- C is a set of *operator-counting constraints*,
- Operator-counting heuristic is defined by the objective value of the linear program under constraint set C.

Operator-Counting Constraints

Operator-Counting Variables

$Count_o$ for each variable $o \in O$

Operator-Counting Constraint

A linear inequality over operator-counting variables.

Single condition: Every plan must represent a feasible solution for operator-counting constraint c!

Post-Hoc Optimization Constraints (Pommerening et al., 2013)

Post-Hoc Optimization Constraint

$$\sum_{o \in O \setminus N} cost(o) \cdot Count_o \geq h(s)$$

- h: admissible heuristic
- N: set of *non-contributing operators*

Post-hoc optimization constraints are operator-counting constraints (Pommerening et al., 2014).

Non-Contributing Operators

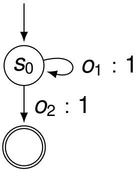
Non-Contributing Operator

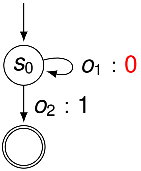
$N \subseteq O$ is a set of *non-contributing operators* if $h(s, cost)$ is an admissible estimate in the planning task with a cost function $cost'$ where $cost'(o) = 0$ for all $o \in N$, or formally

$$h(s, cost) \leq h^*(s, cost').$$

Non-Contributing Operators: Example

$h = |\pi^*|$ for both tasks





$h(s_0, cost) = 1$

$h(s_0, cost) = 1$

estimate still admissible!

Cost-Altered Post-Hoc Optimization Constraints

Cost-Altered Post-Hoc Optimization Constraint

introduce alternative cost function *cost'*:

$$\sum_{o \in O \setminus N} cost'(o) \cdot Count_o \geq h(s, cost')$$

- *h*: admissible heuristic under cost function *cost'*,
- *N*: set of *non-contributing operators*

Cost-Altered Post-Hoc Optimization Constraints

Proposition

Cost-altered post-hoc optimization constraints are operator-counting constraints.

Proof Sketch

Let

- π: plan for Π,
- π_R: same plan with non-contributing operators are removed
- π and π_R have the same plan cost under $cost''$.

Proof Sketch

Post-Hoc Optimization constraint under *cost′*:

$$\sum_{o \in O \setminus N} cost'(o) \cdot Count_o \quad \overset{?}{\geq} \quad h(s, cost')$$

Let π be a plan. We plug in the variable assignment represented by the plan π, e.g.

$$Count_o = occur(o, \pi).$$

Proof Sketch

1. We introduce a cost function

$$cost''(o) = \begin{cases} 0 & \text{if } o \in N, \\ cost'(o) & \text{otherwise.} \end{cases}$$

transform left-hand side to $cost''$: corresponds to reduced "plan" π_R under $cost''$.

$$\sum_{o \in O \setminus N} cost'(o) \cdot occur(o, \pi) \overset{?}{\geq} h(s, cost')$$

$$\parallel$$

$$\sum_{o \in O \setminus N} cost''(o) \cdot occur(o, \pi)$$

Proof Sketch

2. reintroduce non-contributing operators again. Corresponds to plan π under *cost''*.

$$cost''(o) = \begin{cases} 0 & \text{if } o \in N, \\ cost'(o) & \text{otherwise.} \end{cases}$$

$$\sum_{o \in O \setminus N} cost'(o) \cdot occur(o, \pi) \overset{?}{\geq} h(s, cost')$$

$$\parallel$$

$$\sum_{o \in O \setminus N} cost''(o) \cdot occur(o, \pi)$$

$$\parallel$$

$$\sum_{o \in O} cost''(o) \cdot occur(o, \pi)$$

Proof Sketch

2. reintroduce non-contributing operators again. Corresponds to plan π under *cost''*.

$$\sum_{o \in O \setminus N} cost'(o) \cdot occur(o, \pi) \overset{?}{\geq} h(s, cost')$$

$$\|$$

$$\sum_{o \in O \setminus N} cost''(o) \cdot occur(o, \pi)$$

$$\|$$

$$\sum_{o \in O} cost''(o) \cdot occur(o, \pi) \geq h^*(s, cost'')$$

Proof Sketch

3. under the assumption that
   - i. h is admissible under $cost'$ and $cost''$, and
   - ii. N is a set of non-contributing operators

$$
\begin{array}{ccc}
\sum_{o \in O \setminus N} cost'(o) \cdot occur(o, \pi) & \geq & h(s, cost') \\
\parallel & & \\
\sum_{o \in O \setminus N} cost''(o) \cdot occur(o, \pi) & & \wedge\!| \\
\parallel & & \\
\sum_{o \in O} cost''(o) \cdot occur(o, \pi) & \geq & h^{*}(s, cost'')
\end{array}
$$

Cost-Altered Post-Hoc Optimization Constraints

Caveats

- Heuristic h must be admissible under $cost'$ (and $cost''$)
- better, but not guaranteed for all heuristics: admissible under all cost functions!
  - e.g. Pattern Database Heuristics (Edelkamp, 2001)
- Possibility of improved heuristic estimate only when
  - optimal solution under original cost is not a plan,
  - at least one operator has a smaller cost under the altered cost function
- $cost(o) = 0$: operator o has no influence anymore, loss of heuristic information.

Toy Example

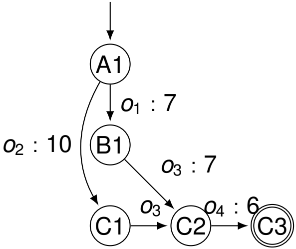


Figure: Transition system $\mathcal{T}$ of planning task Π with variables a and b.
$dom(a) = \{A, B, C\}$,
$dom(b) = \{1, 2, 3\}$

We will use *atomic projections*: abstraction onto single variable.
h : Cost of an optimal plan in the atomic projection
$\Rightarrow$ Pattern Database Heuristic (Edelkamp, 2001)

Toy Example

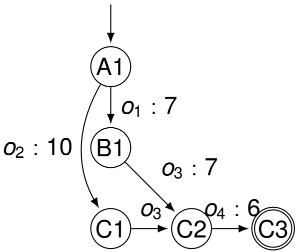


Figure: transition system $\mathcal{T}$ of planning task Π with variables a and b.
$dom(a) = \{A, B, C\}$,
$dom(b) = \{1, 2, 3\}$

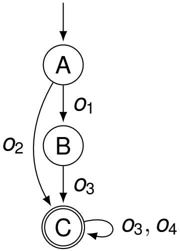


Figure: atomic projection $\mathcal{T}^{\{a\}}$.

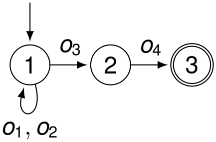


Figure: atomic projection $\mathcal{T}^{\{b\}}$.

Toy Example

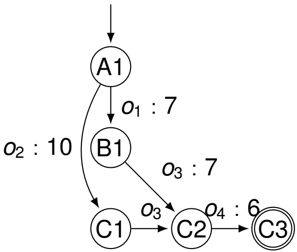


Figure: transition system $\mathcal{T}$ of planning task Π with variables a and b.
$dom(a) = \{A, B, C\}$,
$dom(b) = \{1, 2, 3\}$

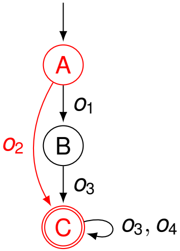


Figure: atomic projection $\mathcal{T}^{\{a\}}$.

$h^{\{a\}}(s_0) = 10$

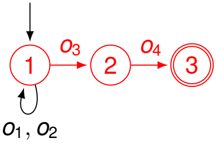
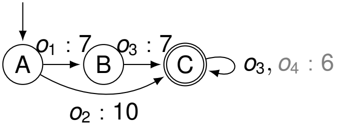
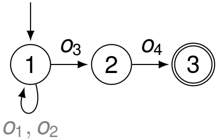


Figure: atomic projection $\mathcal{T}^{\{b\}}$.

$h^{\{b\}}(s_0) = 13$

Toy Example



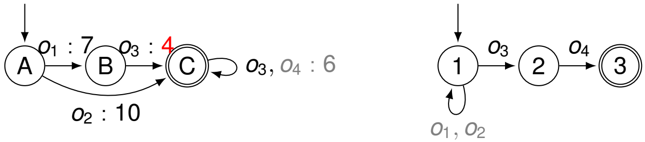


minimize $\sum_{o \in O} cost(o) \cdot Count_0$ subject to

$$\begin{aligned} 7 \cdot Count_{o_1} + 10 \cdot Count_{o_2} \quad +7 \cdot Count_{o_3} \quad &\geq 10 \\ 7 \cdot Count_{o_3} + 6 \cdot Count_{o_4} \quad &\geq 13 \end{aligned}$$

$\Rightarrow h^{\text{LP}}(s_0) = 14$ with solution $Count_{o_3} = 2$.

Toy Example



minimize $\sum_{o \in O} cost(o) \cdot Count_0$ subject to

$$7 \cdot Count_{o_1} + 10 \cdot Count_{o_2} + 4 \cdot Count_{o_3} \geq 10$$

$$4 \cdot Count_{o_3} + 6 \cdot Count_{o_4} \geq 10$$

$\Rightarrow h^{\mathrm{LP}}(s_0) = 20$ with solution $Count_{o_1} = 1, Count_{o_3} = 1, Count_{o_4} = 1$.
Improved heuristic estimate compared to regular post-hoc optimization constraints!

Experiment Setup

- Implemented cost-altering for post-hoc optimization constraints in Fast Downward (Helmert, 2011).
- appropriate subset of planning task from benchmark selection
- Tested implementation on sciCORE grid.

Experiment Setup

Constraint sets tested:

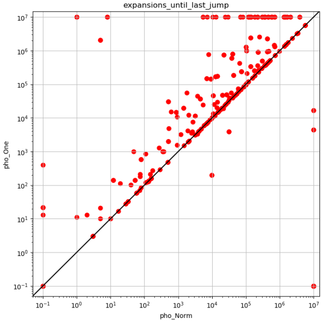
- **SEQ** lower-bound net change constraints
- **LMC** landmark constraints
- **PhO_Norm** regular pattern database constraints
- **PhO_One** cost-altered pattern database constraints with the cost function cost(o) = 1 for all operators
- **PhO_Rand** cost-altered pattern database constraints where the altered cost function assigns each operator a random cost between 1 and its original cost

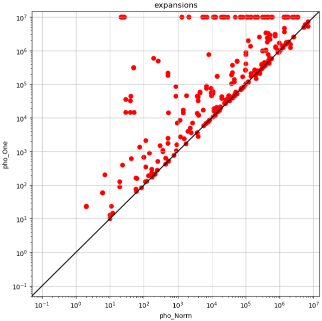
plus combinations thereof

Experiment Results

Cost-altering reduced coverage:

Coverage	PhO_Norm	PhO_One	PhO_Rand	LMC	SEQ
Sum (697)	312	276	244	361	320





Experiment Results: Interpretation

- Only in domains *scanalyzer* and *tetris* was improved initial h-value achieved.
  - domains characterized by loops with near-similar cost
- Otherwise, slight loss of coverage or significant loss in case of PhO_Rand.
- No significant positive or negative interactions on combinations

Conclusion

- Cost-altered post-hoc optimization constraints are operator-counting constraints, but: chance of reaching an improved solution in practice is low
- ⇒ need more informed method for generating alternative cost functions
  - Problem: what is a “good” cost function? Need some kind of objective criterion.
  - ⇒ find cost function that maximises heuristic value while staying admissible, similar to optimal cost partitioning.
    - infeasible in practice?
  - ⇒ something similar to saturated cost partitioning (Seipp and Helmert, 2014)
    - What is criterion for handing out costs?

Thank you for your attention!

Conclusion

- Cost-altered post-hoc optimization constraints are operator-counting constraints, but: chance of reaching an improved solution in practice is low
- $\Rightarrow$ need more informed method for generating alternative cost functions
  - Problem: what is a “good” cost function? Need some kind of objective criterion.
  - $\Rightarrow$ find cost function that maximises heuristic value while staying admissible, similar to optimal cost partitioning.
    - infeasible in practice?
  - $\Rightarrow$ something similar to saturated cost partitioning (Seipp and Helmert, 2014)
    - What is criterion for handing out costs?

Bibliography I

Blai Bonet. An admissible heuristic for SAS$^+$ planning obtained from the state equation. In *Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence*, pages 2268–2274, 2013.

Stefan Edelkamp. Planning with pattern databases. In *Proceedings of the Sixth European Conference on Planning*, pages 84–90, 2001.

Malte Helmert. The Fast Downward planning system. *Journal of Artificial Intelligence Research*, 26:191–246, 2011.

Michael Katz and Carmel Domshlak. Optimal admissible composition of abstraction heuristics. *Artificial Intelligence*, 174(12-13): 767–798, 2010.

Florian Pommerening, Gabriele Röger, and Malte Helmert. Getting the most out of pattern databases for classical planning. In *Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence*, pages 2357–2364, 2013.

Bibliography II

Florian Pommerening, Gabriele Röger, Malte Helmert, and Blai Bonet. LP-based heuristics for cost-optimal planning. In *Proceedings of the Twenty-Fourth International Conference on Automated Planning and Scheduling*, pages 226–234, 2014.

Jendrik Seipp and Malte Helmert. Diverse and additive cartesian abstraction heuristics. In *Proceedings of the Twenty-Fourth International Conference on Automated Planning and Scheduling*, pages 289–297, 2014.

Lin Zhu and Robert Givan. Landmark extraction via planning graph propagation. In *Printed Notes of International Conference on Automated Planning and Scheduling 2003 Doctoral Consortium*, 2003.