Delete-Relaxation Heuristics for Lifted Classical Planning

Augusto B. Corrêa, Guillem Francès,
Florian Pommerening, and Malte Helmert.

University of Basel, Switzerland
Universitat Pompeu Fabra, Spain

```
(:action example
 :parameters (?X ?Y)
 :precondition (and (P ?X ?Y)
                    (R ?X))
 :effect (and (not (P ?X ?Y))
              (Q ?X)
              (R ?Y)))
```

```
(:action example
 :parameters (?X ?Y)
 :precondition (and (P ?X ?Y)
                    (R ?X))
 :effect (and (not (P ?X ?Y))
              (Q ?X)
              (R ?Y)))
```

```
(:action example
 :parameters (?X ?Y)
 :precondition (and (P ?X ?Y)
                    (R ?X))
 :effect (and (not (P ?X ?Y))
              (Q ?X)
              (R ?Y)))
```

$$R(Y) \leftarrow P(X, Y), R(X).$$

$$\overbrace{\underbrace{R(Y)}_{\text{head}} \leftarrow \underbrace{P(X,Y), R(X)}_{\text{body}}}^{\text{rule}}.$$

$$R(Y) \leftarrow P(X, Y), R(X).$$

$$s \coloneqq \{P(0,1); P(1,2); R(0)\}.$$

$$R(Y) \leftarrow P(X, Y), R(X).$$

$$s := \{P(0, 1); P(1, 2); R(0)\}.$$

$$R(1) \leftarrow P(0, 1), R(0).$$

$$R(Y) \leftarrow P(X, Y), R(X).$$

$$s \coloneqq \{P(0,1); P(1,2); R(0)\}.$$

$$R(1) \leftarrow P(0,1), R(0).$$

$$R(2) \leftarrow P(1,2), R(1).$$

```
(:action example
 :parameters (?X ?Y)
 :precondition (and (P ?X ?Y)
                    (R ?X))
 :effect (and (not (P ?X ?Y))
              (Q ?X)
              (R ?Y)))
```

```
(:action example
 :parameters (?X ?Y)
 :precondition (and (P ?X ?Y)
                    (R ?X))
 :effect (and (not (P ?X ?Y))
              (Q ?X)
              (R ?Y)))
```

$$R(Y) \leftarrow P(X,Y), R(X).$$

$$Q(X) \leftarrow P(X,Y), R(X).$$

```
(:goal (R 1) (Q 0))
```

```
(:goal (R 1) (Q 0))
```

$$R(Y) \leftarrow P(X, Y), R(X).$$

$$Q(X) \leftarrow P(X, Y), R(X).$$

$$goal() \leftarrow R(1), Q(0).$$

$$R(Y) \leftarrow P(X, Y), R(X).$$

$$Q(X) \leftarrow P(X, Y), R(X).$$

$$goal() \leftarrow R(1), Q(0).$$

$$s \coloneqq \{P(0,1); P(1,2); R(0)\}.$$

$$R(Y) \leftarrow P(X, Y), R(X).$$

$$Q(X) \leftarrow P(X, Y), R(X).$$

$$goal() \leftarrow R(1), Q(0).$$

$$s \coloneqq \{P(0, 1); P(1, 2); R(0)\}.$$

$$R(1) \leftarrow P(0, 1), R(0).$$

$$R(2) \leftarrow P(1, 2), R(1).$$

$$R(Y) \leftarrow P(X, Y), R(X).$$

$$Q(X) \leftarrow P(X, Y), R(X).$$

$$goal() \leftarrow R(1), Q(0).$$

$$s \coloneqq \{P(0,1); P(1,2); R(0)\}.$$

$$R(1) \leftarrow P(0, 1), R(0).$$

$$R(2) \leftarrow P(1, 2), R(1).$$

$$Q(0) \leftarrow P(0, 1), R(0).$$

$$Q(1) \leftarrow P(1, 2), R(1).$$

$$goal() \leftarrow R(1), Q(0).$$

$$R(Y) \leftarrow P(X, Y), R(X).$$

$$Q(X) \leftarrow P(X, Y), R(X).$$

$$goal() \leftarrow R(1), Q(0).$$

$$s \coloneqq \{P(0,1); P(1,2); R(0)\}.$$

$$R(1) \leftarrow P(0,1), R(0).$$

$$R(2) \leftarrow P(1,2), R(1).$$

$$Q(0) \leftarrow P(0,1), R(0).$$

$$Q(1) \leftarrow P(1,2), R(1).$$

$$goal() \leftarrow R(1), Q(0).$$

How to compute h^{add} ***from the relaxed reachable atoms?***

Ground planning:

$$h^{\mathrm{add}}(s) = \sum_{g \in goal} h^{\mathrm{add}}(g)$$

$$h^{\mathrm{add}}(p) = \begin{cases} 0, & \text{if } p \in s \\ \min_{a \in A(p)} \left\{ cost(a) + \sum_{q \in pre(a)} h^{\mathrm{add}}(q) \right\}, & \text{otherwise,} \end{cases}$$

where $A(p)$ is the set of actions with p in the effect

Lifted planning:

$$h^{\text{add}}(s) = \sum_{g \in goal} h^{\text{add}}(g)$$

$$h^{\text{add}}(p) = \begin{cases} 0, & \text{if } p \in s \\ \min\limits_{r \in A(p)} \left\{ cost(r) + \sum\limits_{q \in body(r)} h^{\text{add}}(q) \right\}, & \text{otherwise,} \end{cases}$$

where $A(p)$ is the set of rules with p in the head

Do we need to compute all relaxed reachable atoms in advance?

$$s \coloneqq \{P(0,1); P(1,2); R(0)\}\,.$$

$$\begin{aligned} Reached \coloneqq s \cup \{&R(1); R(2);\\ &Q(0); Q(1);\\ &goal()\}. \end{aligned}$$

$$s' \coloneqq \{P(0,1); P(1,2); P(2,3); P(3,4); R(0)\}\,.$$

$$\begin{aligned} \mathit{Reached} \coloneqq s \cup \{ & R(1); R(2); R(3); R(4); \\ & Q(0); Q(1); Q(2); Q(3); \\ & \mathit{goal}()\}. \end{aligned}$$

$$PriorityQueue \coloneqq \langle P(0,1); P(1,2); P(2,3); P(3,4); R(0) \rangle$$

$$Reached \coloneqq \{\}$$

$$PriorityQueue \coloneqq \langle P(1,2); P(2,3); P(3,4); R(0) \rangle$$

$$Reached \coloneqq \{P(0,1)\}$$

Can we ground any new rule with *Reached*?

$$PriorityQueue \coloneqq \langle P(2,3); P(3,4); R(0) \rangle$$

$$Reached \coloneqq \{P(0,1); P(1,2)\}$$

Can we ground any new rule with *Reached*?

$PriorityQueue := \langle\rangle$

$$Reached := \{P(0,1); P(1,2); P(2,3); P(3,4); R(0)\}$$

Can we ground any new rule with $Reached$?

$PriorityQueue := \langle\rangle$

$$Reached := \{P(0,1); P(1,2); P(2,3); P(3,4); R(0)\}$$

Can we ground any new rule with *Reached*?

$$R(1) \leftarrow P(0,1), R(0).$$

$$Q(0) \leftarrow P(0,1), R(0).$$

$$PriorityQueue \coloneqq \langle Q(0); R(1) \rangle$$

$$Reached \coloneqq \{P(0,1); P(1,2); P(2,3); P(3,4); R(0)\}$$

$$R(1) \leftarrow P(0,1), R(0).$$

$$Q(0) \leftarrow P(0,1), R(0).$$

$$PriorityQueue \coloneqq \langle R(1) \rangle$$

$$Reached \coloneqq \{P(0,1); P(1,2); P(2,3); P(3,4); R(0); Q(0)\}$$

Can we ground any new rule with *Reached*?

$$PriorityQueue \coloneqq \langle\rangle$$

$$Reached \coloneqq \{P(0,1); P(1,2); P(2,3); P(3,4); R(0); Q(0); R(1)\}$$

Can we ground any new rule with *Reached*?

$$R(2) \leftarrow P(1,2), R(1).$$

$$Q(1) \leftarrow P(1,2), R(1).$$

$$goal() \leftarrow R(1), Q(0).$$

$$PriorityQueue \coloneqq \langle goal(), Q(1), R(2) \rangle$$

$$Reached \coloneqq \{P(0,1); P(1,2); P(2,3); P(3,4); R(0); Q(0); R(1)\}$$

$$R(2) \leftarrow P(1,2), R(1).$$

$$Q(1) \leftarrow P(1,2), R(1).$$

$$goal() \leftarrow R(1), Q(0).$$

$$PriorityQueue \coloneqq \langle Q(1), R(2) \rangle$$

$$Reached \coloneqq \{P(0,1); \ldots; P(3,4); R(0); Q(0); R(1); goal()\}$$

$$PriorityQueue \coloneqq \langle Q(1), R(2) \rangle$$

$$Reached \coloneqq \{P(0,1); \ldots; P(3,4); R(0); Q(0); R(1); goal()\}$$

When $goal()$ is removed from the queue, we can stop.

We can compute $h^{\mathrm{add}}(s')$ with only 8 relaxed reachable atoms.

Computational Cost

- + Same number of atoms evaluated as ground variant
- + Avoid exponential grounding step
- – Evaluating an atom is more expensive

Lazy Search with Preferred Operators

- $goal()$ is useful; and
- Atom q is useful if $q \in body(r)$ where r is the rule achieving some useful atom p with cheapest cost.
- Operator is preferred if it has an useful atom in its effect.

Does this work in practice?

Comparison to Lifted Planners

	New		Other	
Coverage	h^{add}	h^{add} + PO/DQ	h^{gc}	L-RPG
IPC (1001)	616	**754**	597	331
HTG (418)	251	362	**382**	137
Total (1419)	867	**1116**	979	468

Comparison to Ground Planners

	Lifted		Ground	
Coverage	h^{add}	h^{add}+ PO/DQ	h^{add}	h^{add}+ PO/DQ
IPC (1001)	616	754	755	**839**
HTG (418)	251	**362**	264	298
Total (1419)	867	1116	1019	**1137**

Conclusion

- Lifted h^{add} (and h^{max});
- Preferred operators for lifted tasks;
- Competitive with ground counterpart; and
- Better than other lifted heuristics overall.