Authors:
Erick Rosete Beas — er165@uni-freiburg.de
Jessica Lizeth Borja Diaz — jb986@uni-freiburg.de

# Principles of AI Planning

## Exercise Sheet 11

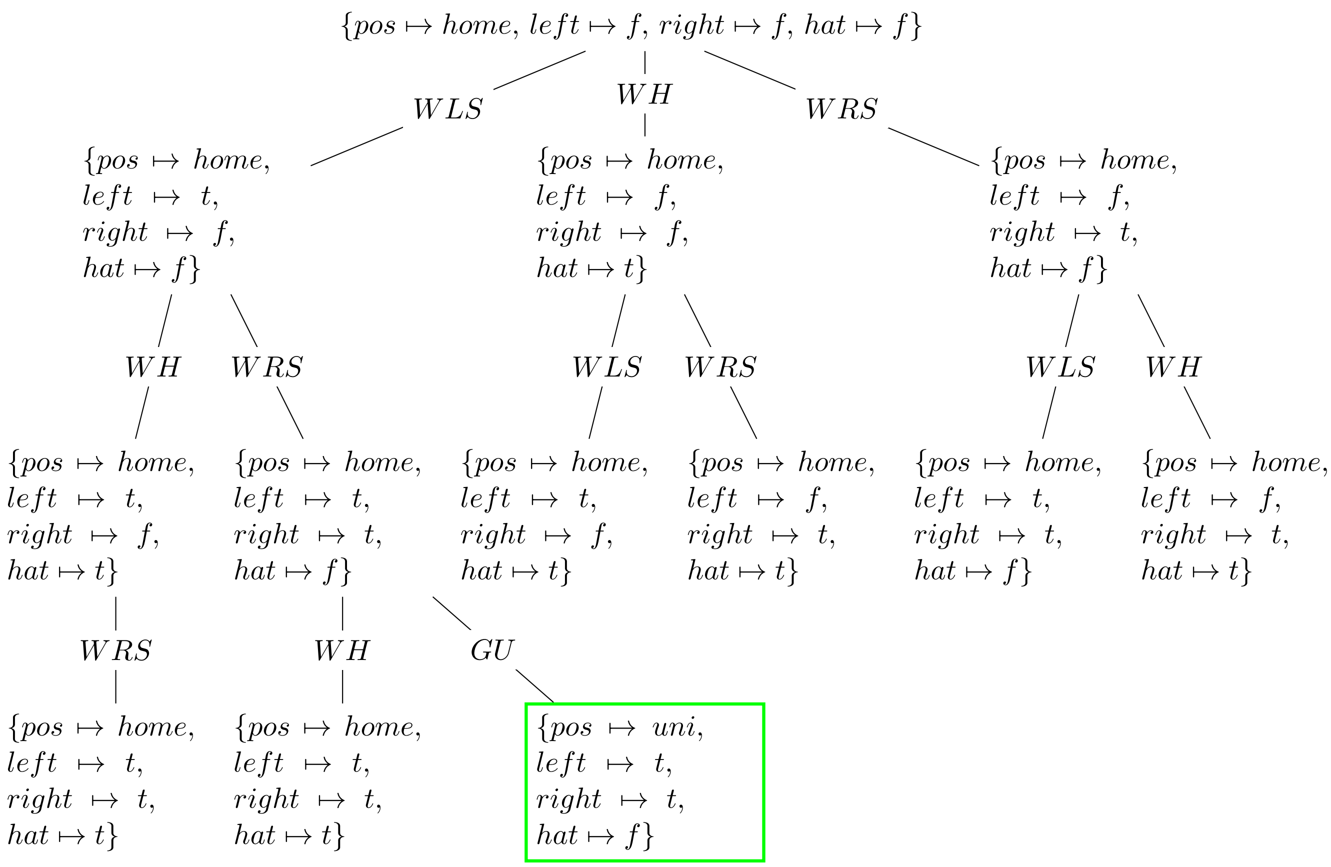
24.01.2020

## Exercise 11.1 - Strong stubborn sets

Consider the $SAS^+$ planning task $\Pi$ with variables $V = \{pos, left, right, hat\}, \mathcal{D}_{pos} = \{home, uni\}$ and $\mathcal{D}_{left} = \mathcal{D}_{right} = \mathcal{D}_{hat} = \{t, f\}$. The initial state $I = \{pos \mapsto home, left \mapsto f, right \mapsto f, hat \mapsto f\}$ and the goal specification is $\gamma = \{pos \mapsto uni\}$. There are four operators in O, namely

$$
\begin{aligned}
\text{wear-left-shoe (WLS)} &= \langle pos = home \wedge left = f, left := t\rangle \\
\text{wear-right-shoe (WRS)} &= \langle pos = home \wedge right = f, right := t\rangle \\
\text{wear-hat (WH)} &= \langle pos = home \wedge hat = f, hat := t\rangle \\
\text{go-to-university (GU)} &= \langle pos = home \wedge left = t \wedge right = t, pos := uni\rangle
\end{aligned}
$$

(a) Draw the breadth-first search graph (with duplicate detection) for planning task $\Pi$ without any form of partial-order reduction.



(b) Draw the breadth-first search graph (with duplicate detection) for planning task Π using strong stubborn set pruning. For each expansion of a node for a state $s$, specify in detail how $T_s$ (and thus $T_{app(s)}$) are computed, i.e., explain how the initial disjunctive action landmark is chosen and how operators are iteratively added to $T_s$ as part of necessary enabling sets or interfering operators, respectively. Break ties in favor of *wear-left-shoe* over *wear-right-shoe.*

**How many node expansion do you save with strong stubborn sets compared to plain breadth-first search? What about the lengths of the resulting solutions?**

$$S_1 = \{pos \mapsto home,\, left \mapsto f,\, right \mapsto f,\, hat \mapsto f\}$$

$$\Big|\; WLS \;\Big|$$

$$S_2 = \{pos \mapsto home,\, left \mapsto t,\, right \mapsto f,\, hat \mapsto f\}$$

$$\Big|\; WRS \;\Big|$$

$$S_3 = \{pos \mapsto home,\, left \mapsto t,\, right \mapsto t,\, hat \mapsto f\}$$

$$\Big|\; GU \;\Big|$$

$$\boxed{\{pos \mapsto uni,\, left \mapsto t,\, right \mapsto t,\, hat \mapsto f\}}$$

The strong stubborn set in $S_1$ is determined by taking into account the disjunctive action landmark, we know the operator *go-to-university* must be inside this set.
As *go-to-university* is not applicable in the initial state, then we need to add a neccessary enabling set for this operator, a possible NES is $\{wear\text{-}left\text{-}shoe\}$ or $\{wear\text{-}right\text{-}shoe\}$, we broke ties by adding the operator *wear-left-shoe* to the strong stubborn set.
As *wear-left-shoe* is applicable, we must add all operators that interfere, the only operator that interferes with this operator is *go-to-university* but it is already in the set, therefore:

$$T_{S_1} = \{go\text{-}to\text{-}university, wear\text{-}left\text{-}shoe\}$$
$$T_{S_1} \cap T_{app(S_1)} = \{wear\text{-}left\text{-}shoe\}$$

The strong stubborn set in $S_2$ is determined using again $\{go\text{-}to\text{-}university\}$ as DAL.
As *go-to-university* is not applicable in $S_2$, then the smallest NES is $\{wear\text{-}right\text{-}shoe\}$
As *wear-right-shoe* is applicable, we must add the operator *go-to-university* as it interferes, but it is already in the set, therefore:

$$T_{S_2} = \{go\text{-}to\text{-}university, wear\text{-}right\text{-}shoe\}$$
$$T_{S_2} \cap T_{app(S_2)} = \{wear\text{-}right\text{-}shoe\}$$

The strong stubborn set in $S_3$ is determined using again $\{go\text{-}to\text{-}university\}$ as DAL.
As *go-to-university* is applicable in $S_3$, then we need to add all operators that interfere, in this case *go-to-university* disables $\{wear\text{-}left\text{-}shoe, wear\text{-}right\text{-}shoe, wear\text{-}hat\}$

$$T_{S_3} = \{go\text{-}to\text{-}university, wear\text{-}left\text{-}shoe, wear\text{-}right\text{-}shoe, wear\text{-}hat\}$$
$$T_{S_3} \cap T_{app(S_3)} = \{go\text{-}to\text{-}university, wear\text{-}hat\}$$

In this case the first visited node was a goal state, after that we stopped the search.

By prunning with the strong stubborn set we only visited 4 states, in contrast without prunning we visited 13 states, 9 states more, which represents approximately a 30% the previous graph.

## Exercise 11.2 - Weak vs. strong stubborn sets

Show that *weak* stubborn sets admint exponentially more pruning than *strong* stubborn sets.