Principles of AI Planning

Prof. Dr. B. Nebel, R. Mattmüller
N. Mayer
Winter Semester 2012/2013

University of Freiburg
Department of Computer Science

Exercise Sheet 5

Due: November 30th, 2012

Exercise 5.1 (h^+ heuristic, 2+2 points)
A 15-puzzle planning task $\Pi = \langle A, I, O, \gamma \rangle$ is given as

$$
\begin{aligned}
A &= \{\mathit{empty}(p_{i,j}) \mid 0 \le i, j \le 3\} \cup \{\mathit{at}(t_k, p_{i,j}) \mid 0 \le i, j \le 3, 0 \le k \le 14\}, \\
O &= \{\mathit{move}(t_m, p_{i,j}, p_{k,l}) \mid 0 \le i, j, k, l \le 3, 0 \le m \le 14, \\
&\quad (i = k \text{ and } |j - l| = 1) \text{ or } (j = l \text{ and } |i - k| = 1)\}, \\
\gamma &= \bigwedge_{0 \le m \le 14} \mathit{at}(t_m, p_{\lfloor m/4 \rfloor, m\%4})
\end{aligned}
$$

Action $\mathit{move}(t_m, p_{i,j}, p_{k,l})$ moves tile t_m from position $p_{i,j}$ to position $p_{k,l}$:

$$
\begin{aligned}
\mathit{move}(t_m, p_{i,j}, p_{k,l}) = \langle &\mathit{at}(t_m, p_{i,j}) \wedge \mathit{empty}(p_{k,l}), \\
&\mathit{at}(t_m, p_{k,l}) \wedge \mathit{empty}(p_{i,j}) \wedge \neg \mathit{at}(t_m, p_{i,j}) \wedge \neg \mathit{empty}(p_{k,l}) \rangle
\end{aligned}
$$

A syntactically possible state is *legal* if each tile t_m is at some position p_{ij}, if no two tiles are at the same position and if the remaining position is the only one that is *empty*. The initial state is an arbitrary state that is legal.

One possible heuristic for the 15-puzzle is the Manhattan-distance heuristic $h^{Manhattan}$: It sums the Manhattan distances of all tiles from their current positions to their target positions, where the Manhattan distance between position $p_{i,j}$ and $p_{k,l}$ is given as $|i - k| + |j - l|$.
The h^+ heuristic estimates the distance of state s to the closest goal state as the length of the optimal plan in the relaxed planning task (with initial state s).

(a) Show that $h^+(s) \ge h^{Manhattan}(s)$ for each legal state s of a 15-puzzle planning task.

(b) Show that $h^+(s) > h^{Manhattan}(s)$ for at least one state s of a 15-puzzle planning task.

Exercise 5.2 (PNF and relaxation, 1.5+1+1.5 Points)
Consider the following (possibly slightly familiar) planning task $\Pi = \langle A, I, O, \gamma \rangle$ with

$$
\begin{aligned}
A &= \{\text{visible}, \text{at-party}, \text{guests-angry}\}, \\
I &= \{\text{visible} \mapsto 1, \text{at-party} \mapsto 1, \text{guests-angry} \mapsto 0\} \\
O &= \{\text{toggle-ring}, \text{leave-party}\}, \\
\text{toggle-ring} &= \langle \top, (\text{visible} \rhd \neg\text{visible}) \wedge (\neg\text{visible} \rhd \text{visible}) \rangle, \\
\text{leave-party} &= \langle \text{at-party}, \neg\text{at-party} \wedge (\text{visible} \rhd \text{guests-angry}) \rangle \\
\gamma &= \neg\text{at-party} \wedge \neg\text{guests-angry}.
\end{aligned}
$$

(a) Give the PNF of the task planning task Π. You don't have to step through the transformation algorithm, just write down the result. Watch for loss of ENF!

(b) Give the relaxation Π^+ of Π'.

(c) Give a sequence π of operators (as short as possible) from O' such that π is *not* a plan of Π', but π^+ is a plan of Π^+.

Exercise 5.3 (Polynomial Planning, 2 Points)

For a fixed number $g \in \mathbb{N}$, consider STRIPS planning with negative preconditions (and possibly goal literals) under the restriction that the goal formula γ is a conjunction of at most g literals and that each operator precondition is a single literal (positive or negative). There are no conditional effects. I.e., each operator has the form $\langle \ell, a_1 \wedge \cdots \wedge a_n \wedge \neg d_{n+1} \wedge \cdots \wedge \neg d_{n+m} \rangle$, ℓ being a positive or negative literal. Let $\textsc{PlanEx}^1_g$ be the problem of deciding, for a given planning task Π in the form given above, if there exists a plan for Π.
Show: $\textsc{PlanEx}^1_g \in \mathbf{P}$, i.e., there exists a polynomial-time algorithm that decides $\textsc{PlanEx}^1_g$.
Hint: Regression search, representing (sub-)goals as sets of literals that never grow during regression steps.

Note: The exercise sheets may and should be worked on in groups of two students. Please state both names on your solution (this also holds for submissions by e-mail).