

Comp3620/Comp6320 Artificial Intelligence

Tutorial 6: Heuristics, Regression, and Partial-Order Planning

May 15-18, 2018

Exercise 1 (Delete Relaxation)

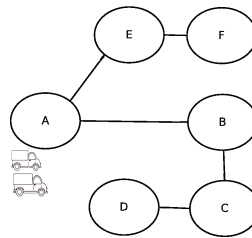


Figure 1: Delivering Problem

Consider the following delivery problem. Two driverless trucks (T1 and T2) can autonomously deliver the products that customers have requested. The products are initially at the depot (location A), and our trucks can reach the customers identified by their locations (B,C,D,E,F) by following suitable paths connecting adjacent locations. Just two out of these five customers (F and D) have to be served. Assume for simplicity that the two trucks have always enough products on-board. Ignore any cost that could be associated to the actions. A typed STRIPS description of the problem is as follows:

- Types: Locations = {A,B,C,D,E,F}, Trucks = {T1,T2}
- Propositions:
 - $\text{at}(t,x)$, $x \in \text{Locations}$, $t \in \text{Trucks}$
 - $\text{delivered}(x)$ $x \in \text{Locations}$
 - $\text{adjacent}(x,y)$, $x, y \in \text{Locations}$
- Actions:
 - $\text{go}(t,x,y)$, $t \in \text{Trucks}$, $x, y \in \text{Locations}$
 - * preconditions: $\{\text{at}(t,x), \text{adjacent}(x,y)\}$
 - * add-list: $\{\text{at}(t,y)\}$
 - * del-list: $\{\text{at}(t,x)\}$
 - $\text{deliver}(t,x)$, $t \in \text{Trucks}$, $x \in \text{Locations}$
 - * preconditions: $\{\text{at}(t,x)\}$
 - * add-list: $\{\text{delivered}(x)\}$
 - * del-list: \emptyset

- Initial state: $\{at(T1,A),at(T2,A)\} \cup \{ adjacent(x,y) \mid x,y \in \text{Locations, there is an edge between } x \text{ and } y \text{ in the graph of Figure1} \}$
- Goal: $\{delivered(F),delivered(D),at(T1,A),at(T2,A)\}$

Questions:

1. Explain the concepts of the delete relaxation of a problem P and of a relaxed-plan for P .
2. Write the optimal relaxed plan for this problem. What are the values of h^+ and h^* at the initial state?
3. Write a partially ordered plan and a parallel plan solving the problem.

Question 1:

The delete relaxation P^+ of a problem the same problem as P where all actions have been transformed by removing their delete list. A relaxed plan is a (not necessarily optimal) solution to the delete-relaxation P^+ .

As an aside: P^+ is simple problem than P hence an optimal relaxed plan cost underestimates the true cost and can be used as an admissible heuristic. One issue is that P^+ is NP-hard to solve optimally. Therefore, many approaches further relax the problem to get an admissible heuristics (e.g. the admissible hmax) or compute non-optimal relaxed plans and non-admissible heuristics (e.g. the FF heuristic).

Question 2:

The easiest way to answer the question is to recall that, in the delete-relaxation, once a proposition becomes true, then it remains true forever, and that an action that was once applicable remains applicable for ever. In the present case, a relaxed plan can assume that a truck is at a location that it already visited, in particular, location A. So that as soon as a truck has delivered the first item (for instance at location D), it can assume it is back in A.

- Optimal Relaxed Plan: $\langle go(T1,A,B), go(T1,B,C), go(T1,C,D), deliver(T1,D), go(T2,A,E), go(T2,E,F), deliver(T2,F) \rangle$.
- $h^+(s_0) = 7$
- Note: The same plan with T2 replaced with T1 would still be a relaxed plan since when T1 is back in A, it can go on delivering to F.
- Optimal Sequential Plan: $\langle go(T1,A,B), go(T1,B,C), go(T1,C,D), deliver(T1,D), go(T1,D,C), go(T1,C,B), go(T1,B,A), go(T2,A,E), go(T2,E,F), deliver(T2,F), go(T2,F,E), go(T2,E,A) \rangle$.
- $h^*(s_0) = 12$

Question 3:

- Partial Order Plan: in this case, it corresponds to the two trucks operating asynchronously
 $\langle \{ go(T1,A,B), go(T1,B,C), go(T1,C,D), deliver(T1,D), go(T1,D,C), go(T1,C,B), go(T1,B,A), go(T2,A,E), go(T2,E,F), deliver(T2,F), go(T2,F,E), go(T2,E,A) \},$
 $\{ go(T1,A,B) \prec go(T1,B,C),$
 $go(T1,B,C) \prec go(T1,C,D),$
 $go(T1,C,D) \prec deliver(T1,D),$
 $deliver(T1,D) \prec go(T1,D,C),$
 $go(T1,D,C) \prec go(T1,C,B),$
 $go(T1,C,B) \prec go(T1,B,A),$
 $go(T2,A,E) \prec go(T2,E,F),$
 $go(T2,E,F) \prec deliver(T2,F),$
 $deliver(T2,F) \prec go(T2,F,E),$
 $go(T2,F,E) \prec go(T2,E,A) \} \rangle$

- Parallel Plan: there are many possibilities, for instance
 $\langle \{ \text{go}(\text{T1}, \text{A}, \text{B}), \text{go}(\text{T2}, \text{A}, \text{E}) \},$
 $\{ \text{go}(\text{T1}, \text{B}, \text{C}), \text{go}(\text{T2}, \text{E}, \text{F}) \},$
 $\{ \text{go}(\text{T1}, \text{C}, \text{D}), \text{deliver}(\text{T2}, \text{F}) \},$
 $\{ \text{deliver}(\text{T1}, \text{D}) \},$
 $\{ \text{go}(\text{T1}, \text{D}, \text{C}), \text{go}(\text{T2}, \text{F}, \text{E}) \},$
 $\{ \text{go}(\text{T1}, \text{C}, \text{B}), \text{go}(\text{T2}, \text{E}, \text{A}) \},$
 $\{ \text{go}(\text{T1}, \text{B}, \text{A}) \} \rangle$

Exercise 2 (Regression Planning)

Consider a small propositional STRIPS planning problem with a set of propositions $P = \{p, q, r, s\}$, and actions $a_1 \dots a_6$. The preconditions and effects of the actions are described in the following table:

Action	PRE	EFF ⁺	EFF ⁻	Action	PRE	EFF ⁺	EFF ⁻
a_1	$\{p, q\}$	$\{r\}$	$\{p\}$	a_4	$\{r\}$	$\{s\}$	$\{q\}$
a_2	$\{q\}$	$\{p\}$	$\{q\}$	a_5	$\{r, s\}$	$\{q\}$	$\{r\}$
a_3	$\{p, q\}$	$\{s\}$	$\{q\}$	a_6	$\{s\}$	$\{r\}$	$\{s\}$

Questions:

1. Let g be a goal in regression (backward search) planning, and a be a *propositional* STRIPS action such that $a = \langle \text{PRE}(a), \text{EFF}^+(a), \text{EFF}^-(a) \rangle$. State the condition under which a is relevant to g and explain how to compute the regression of g through a .
2. For each of the 6 actions described above, state whether the goal $\{r, s\}$ can be regressed through that action, and if yes, what the result is.

Question 1:

An action a is relevant for goal g iff:

- $g \cap \text{EFF}^+(a) \neq \emptyset$, i.e. if it achieves at least one of g 's propositions
- $g \cap \text{EFF}^-(a) = \emptyset$, i.e. it does not destroy one of g 's propositions

If a is relevant for g then the regression of g through a is defined and is: $g' = \gamma^{-1}(g, a) = (g \setminus \text{EFF}^+(a)) \cup \text{PRE}(a)$.

Question 2:

- a_1 : yes, $g' = \{p, q, s\}$
- a_2 : no, the action does not achieve a proposition in g
- a_3 : yes, $g' = \{p, q, r\}$
- a_4 : yes, $g' = \{r\}$
- a_5 : no, the action deletes a proposition in g
- a_6 : no, the action deletes a proposition in g

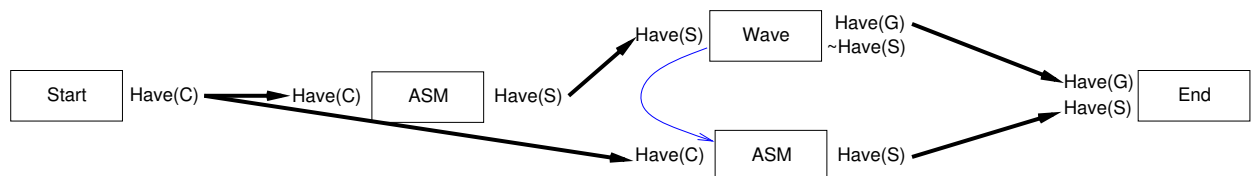
Exercise 3 (Partial-Order Planning)

Examine the following partial plan. As in the previous tutorial, the goal is to have a sheep and a goat. The *ASM* (automated sheep machine) operator yields a sheep if you have an ASM card. The *Wave* operator waves your magic wand to turn a sheep into a goat.



1. Which conditions are open (indicate both the condition and the operator for which it is open).
2. List all threats (indicate both the causal link threatened and the threatening action).
3. State how those threats can be resolved.
4. Draw the final plan that the plan-space planning algorithm would produce. Include all new operators, causal links, and ordering constraints required (except those implied by the causal links).
5. Suppose you are executing the resulting plan. What conditions must be true just prior to executing the *Wave* operator in order for the plan to reach the goal?

1. Have(S) for Wave and Have(C) for ASM
2. The threatening action is Wave and the link threaten is that going from ASM to condition Have(S) of End.
3. only one possibility: order Wave before ASM.
4. the complete plan is:



5. Have(C) and Have(S)