OPPA European Social Fund
Prague & EU: We invest in your future.

PAH Course

Assessed Assignment

Date handed out: 12/04/2011
To be handed in at tutorial: 16/05/2011

1. State Transition Systems: Define a state transition system for a ticket vending machine that accepts 10p, 20p, and 50p coins; the price of a ticket is 60p. Users must insert coins until at least 60p have been inserted. The machine then issues a ticket and returns correct change.

2. Situation Calculus: Go to the course book web-site (http://www.laas.fr/planning/) and download the PDDL specification of the DWR domain defined there. Define the `move` operator as a set of applicability, effect, and frame axioms in the style of the situation calculus.

3. State-Space Search:

a. Go to the course book web-site (http://www.laas.fr/planning/) and download the PDDL specification of the two problems defined there. Draw the initial state for the smaller problem (1 robot/2 locations) as a picture. This should show the topology and location of different objects defined in the problem. Draw a possible goal state for the smaller problem. How many different goal states are there?

b. Some operators may have instances with effects that are complementary, i.e. they contain effects `(P …)` and `(not P …)`. Show an example of such an operator in the DWR domain. What would happen if we used such an action in the DWR domain? Why is this not an issue in the two example problems?

c. What are the applicable actions in the initial state of the smaller DWR problem? What are the applicable actions in the (random) planning problem given in the appendix? Ignore actions with inconsistent (complementary) effects.

d. For the same two problems, which actions are relevant for the respective goals? Note that static relations can be used to constrain the search for relevant actions, and type information where available. Again, ignore actions with inconsistent (complementary) effects.

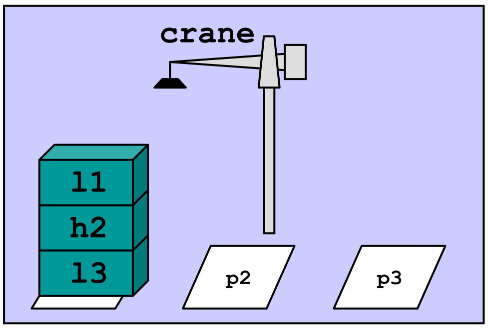
4. Plan-Space Search:

a. Define a partial plan that represents the initial state for the 1 robot/2 locations problem in the plan-space search approach.

b. Simulate the search as performed by the PoP procedure for the first 3 levels. When there is a choice point, choose one option to simulate, but indicate which choice points are deterministic and which are non-deterministic, i.e. which ones are backtrack points.

5. STN/HTN Planning

Suppose the containers in the DWR domain are either heavy or light containers. Ships must be loaded such that the weight distribution is even. To be able to do this more quickly, containers are to be sorted into heavy and light piles. For example, the figure below shows a location where there is one crane, three piles, and three containers. All containers are in the same pile. The light containers are “l1” and “l3”, whereas “h2” is a heavy container.



a. Define a set of STN methods that will perform the task of sorting one given mixed pile into two piles containing only either light or heavy containers each.
b. For the initial state given in the figure above, give a solution plan and show that it accomplishes the task of sorting the stack (according to the definition of an STN solution).
c. Translate your STN methods into HTN methods. What are the most important differences between your two representations?
d. Define a set of state variables that could be used to describe world states for this problem. What would the representation of the state in the figure above look like?

6. Graphplan

The Blocks World can be described in a number of ways. Consider a representation in which there are three operators:

- Move a block from the table onto a block
- Move a block from a block onto the table
- Move a block from a block onto another block

a. Define the Sussman anomaly (see the planning book) as a propositional planning problem. Hint: This will require defining the propositional planning domain.
b. Choose a pair of independent actions that are applicable in the initial state of the Sussman anomaly. What is the result of applying these actions? If there are no such actions, show that there is no such pair.
c. Develop the planning graph for this problem far enough to be able to show an example of a pair of mutex actions that are not dependent. If there are no such actions, stop at proposition layer P_2. Why are there never any mutex but independent actions in the first action layer A_1?

d. What is the time complexity for expanding the planning graph with a new action and a new proposition layer?

e. Graphplan turned out to be significantly faster than any of the algorithms that came before it (state-space search and plan-space search; HTN planners solve a different problem). Why do you think that is?

Appendix: Random Planning Domain and Problem

```
(define (domain random-domain)
  (:requirements :strips)
  (:action op1
    :parameters (?x1 ?x2 ?x3)
    :precondition (and (Q ?x2) (Q ?x1) (R ?x3 ?x3) (R ?x1 ?x1)
      (not (P ?x3 ?x1 ?x3)) (not (S ?x3 ?x1)))
    :effect (and (R ?x1 ?x2) (S ?x1 ?x2) (S ?x2 ?x3) (S ?x1 ?x3)
      (not (R ?x3 ?x3)) (not (R ?x1 ?x1))))
  (:action op2
    :parameters (?x1 ?x2 ?x3)
    :precondition (and (Q ?x3) (R ?x3 ?x2) (P ?x3 ?x1 ?x2) (P ?x3 ?x3 ?x1)
      (not (T ?x1 ?x3 ?x3)) (not (T ?x1 ?x2 ?x3)))
    :effect (and (T ?x2 ?x1 ?x3) (S ?x3 ?x3) (R ?x3 ?x1) (R ?x3 ?x3)
      (not (R ?x3 ?x2))))
  (:action op3
    :parameters (?x1 ?x2 ?x3)
    :precondition (and (T ?x2 ?x3 ?x2) (S ?x2 ?x1) (T ?x1 ?x1 ?x2) (Q ?x1)
      (not (T ?x2 ?x3 ?x1)) (not (S ?x2 ?x3)) (not (R ?x1 ?x3)))
    :effect (and (R ?x2 ?x2) (R ?x1 ?x2) (S ?x2 ?x2) (S ?x2 ?x3)
      (not (S ?x2 ?x1)) (not (T ?x1 ?x1 ?x2)))))


(define (problem random-problem)
  (:domain random-domain)
  (:init
     (Q I) (Q G) (Q H) (Q D) (Q J)
     (P C I G) (P J A E) (P J I E) (P A D C) (P I C C) (P G D C) (P G E I)
      (P B E A) (P B G E) (P I B J) (P I F E) (P C E F) (P G I H) (P J G C)
      (P A A J)
     (T C A F) (T H C E) (T H B I) (T A F H) (T G C J) (T J H A)
     (S D E) (S B J) (S G J) (S J F) (S D G) (S A I) (S A B) (S D I)
     (R I I) (R B C) (R D C) (R I C) (R I F) (R C I))
  (:goal (and (S I G) (S J I))))
```

OPPA European Social Fund
Prague & EU: We invest in your future.