# 15-381: Artificial Intelligence
# Assignment 4: Planning

Due: Tuesday, October 30, 2001 (before class)

This assignment, like the previous one, consists solely of written exercises.

1. Consider a robot domain as shown in Figure 1. The domain consists a house that belongs to Pat, who has a robot-butler. Initially, Pat and the robot are in the living room, and there is an orange in the kitchen. The door between the family room and the kitchen is closed, while all other doors are open. The robot can perform the following actions with the associated costs:

```
go: Go through an open door to an adjacent room; cost: 2.
open-door: Open a door (when in the same room); cost: 3.
pick: Pick an orange (when in the same room); cost: 1.
```

   (a) Consider the task of sending the robot to the kitchen to pick the orange. An $A^*$ search was performed resulting in the search space shown in Figure 2. Choose an admissible heuristic, state it, and fill in the missing values for h and f, so that nodes 1 through 5 were expanded in that order. NOTE: Your heuristic cannot be 0 or some other constant value.

   (b) What is the solution found? Is this an optimal solution? Why or why not?

   (c) Suppose that the robot has executed the plan of Part (b), and now it is in the kitchen with the orange. Consider the task of carrying the orange to the living room (in order to give it to Pat). Show the optimal plan for executing this task. (You do *not* need to show the search space this time, only the resulting plan.)

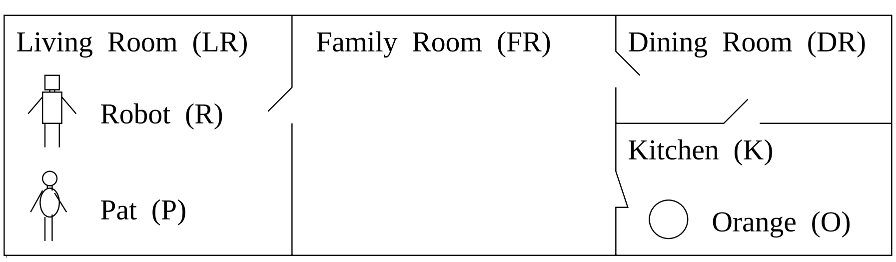


Figure 1: The robot-butler domain.

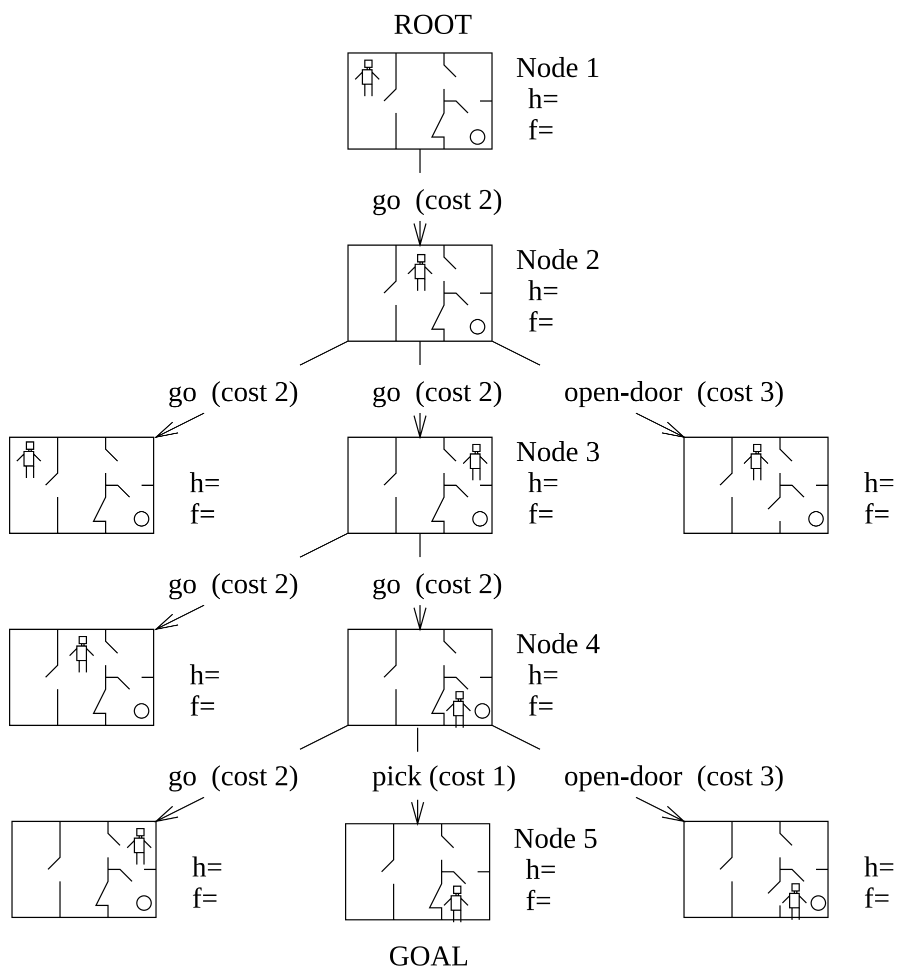


Figure 2: Search space for the robot-butler domain.

(d) Now consider again the initial situation, shown in Figure 1, and suppose that the robot knows *both* goals, i.e. picking up the orange and bringing it to Pat, from the beginning. What is the optimal plan for achieving these two goals?

(e) How does the plan found in (d) differ from the plans found in Parts (b) and (c)? Can you suggest any search algorithm that would find this optimal plan? Is it possible to use $A^*$ (with an admissible heuristic) to perform this search?

(f) Compare the efficiency of searching for the overall optimal plan given the two goals initially, as stated in part (d), with the efficiency of search in Parts (a) and (b). What conclusion can you make about the difficulty of finding the optimal plan for achieving multiple conjunctive goals?

2. Consider the following four operators (cf. page 346 in the textbook):

*Operators*

| | RightShoe | RightSock | LeftShoe | LeftSock |
|---|---|---|---|---|
| *preconds*: | RightSockOn | - | LeftSockOn | - |
| *adds*: | RightShoeOn | RightSockOn | LeftShoeOn | LeftSockOn |
| *deletes*: | - | - | - | - |

Define additional operators for putting a hat and a coat on, respectively, assuming that there are no preconditions for putting on the hat and the coat. Give a partial-order plan that is a solution, and show that there are 180 different linearizations of this solution.

3. The POP algorithm in the textbook is a regression planner, because it adds steps whose effects satisfy unsatisfied conditions in the plan. Progression planners add steps whose preconditions are satisfied by conditions known to be true in the plan. Modify POP so that it works as a progression planner.

4. POP is a nondeterministic algorithm, and has a choice about which operator to add to the plan at each step and how to resolve each threat. Can you think of any domain-independent heuristics for ordering these choices that are likely to improve POP's efficiency?

5. Consider the following operators to load and unload objects into and from containers at and to some locations, to move containers between locations, and to get any new container at any location:

Notice that any list of literals represents a conjunction (e.g., the list of preconditions and the statements of the initial state and the goal).

*Operators*

| | LOAD(o,c,l) | MOVE(c,l1,l2) | UNLOAD(o,c) | GET-NEW(c,l) |
|---|---|---|---|---|
| *preconds*: | At-Container(c,l)<br>Space-Available(c)<br>At-Obj(o,l) | At-Container(c,l1) | Inside (o,c)<br>At-Container(c,l) | - |
| *adds*: | Inside(o,c) | At-Container(c,l2) | At-Obj(o,l) | At-Container(c,l)<br>Space-Available(c) |
| *deletes*: | At-Obj(o,l) | At-Container(c,l1) | Inside(o,c) | - |

(a) Consider the following problem:

Initial State: At-Obj(o1,l1), At-Obj(o2,l1), At-Container(c1,l1), Space-Available(c1).

Goal State: At-Obj(o1,l2), At-Obj(o2,l2).

Pat claims:"There is more than one plan that can solve this problem." Is Pat correct? If yes, then show at least two plans to solve this problem. Otherwise, justify why Pat is incorrect.

(b) Suppose now that Space-Available is not known in the initial state. Consider that you have available an additional operator, CHECK-SPACE, that allows you to check if there is Space-Available. That operator returns Space-Available(c) or ¬ Space-Available(c).

Show a conditional plan to solve the following simple problem:

Initial State: At-Obj(o1,l1), At-Obj(o2,l1), At-Container(c1,l1).

Goal State: At-Obj(o1,l2).

6. Robby is a robot with two grippers—one left and one right. He can pick up and put down balls with the grippers, and each gripper can hold exactly one ball. The following operators represent these actions:

*Operators*

| | Pick(o,r,g) | Drop(o,r,g) |
|---|---|---|
| *preconds*: | At(o,r)<br>At-Robby(r)<br>Free(g) | Carry(o,g)<br>At-Robby(r) |
| *adds*: | Carry(o,g) | At(o,r)<br>Free(g) |
| *deletes*: | At(o,r)<br>Free(g) | Carry(o,g) |

Robby can also move between rooms. This way he can pick up balls in one room and drop them off in another room.

(a) Write an operator `move`, representing the action of moving from one room to another.

Now consider the situation depicted in Figure 3. There are two rooms (`room1`, and `room2`). Robby is in `room1`, and both his grippers are empty. There is only one ball (`ball1`), and it is also in `room1`. The goal is to move `ball1` from `room1` to `room2`.

(b) Specify the Start and Finish operators representing the initial situation and the goal, respectively.

(c) Specify all *consistent* instantiations (with all parameters substituted for atoms) of each of the three operators.

(d) Give a partial-order plan that is a solution to the stated problem. How many linearizations exist for the plan?

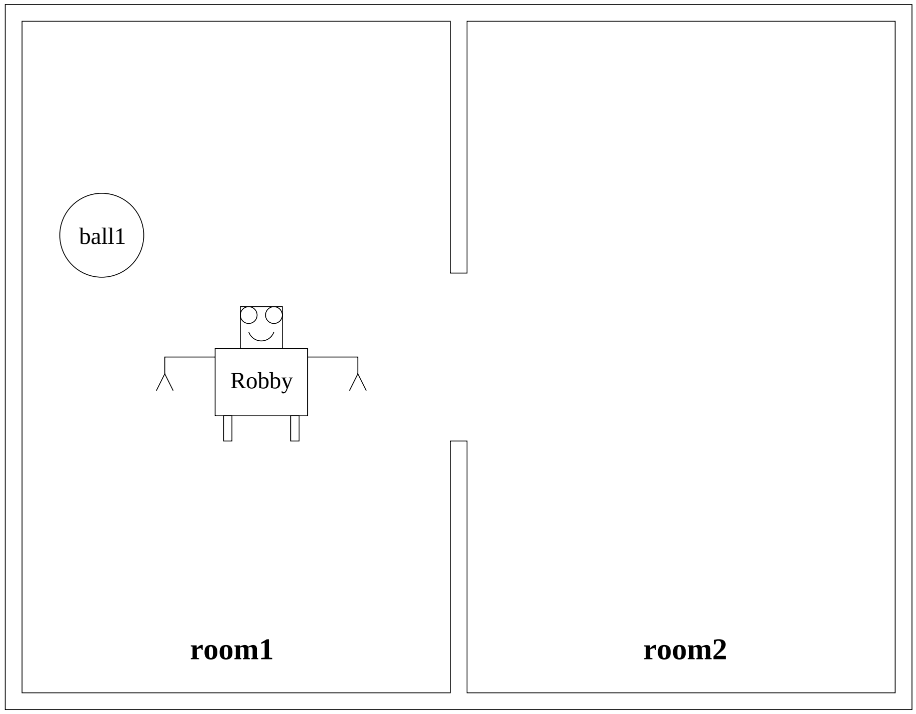


Figure 3: The gripper domain with one ball in room1.

(e) The principle of least-commitment says that a planner should avoid making decisions until there is a good reason to make a choice. Give a least-commitment plan that is a solution to the stated problem. How many fully instantiated plans can be constructed from this plan?