CSIS0270 Artificial Intelligence, 2002–2003
Assignment 4
Deadline May 13, 2003, 5:00pm.

This is a written assignment. Hand-in your answers to the assignment box (R2).

1. ***Planning languages (25%)*** (Adapted from textbook 11.4) The monkey-and-banana problem is faced by a monkey in a laboratory with some bananas hanging out of reach from the ceiling. A box is available that will enable the monkey to reach the bananas if he climbs on it. Initially, the monkey is at *A*, the bananas at *B*, and the box at *C*. The monkey and box have height *Low*, but if the monkey climbs onto the box he will have height *High*, the same as the bananas. The actions available to the monkey include *Go* from one place to another, *Push* an object from one place to another, *ClimbUp* onto or *ClimbDown* from an object, and *Grasp* and *Ungrasp* an object. Grasping results in holding the object if the monkey and object are in the same place at the same height.

   a. Write down the initial state description.

   b. Write down STRIPS-style definition of the six actions.

   c. Suppose the monkey wants to fool the scientists, who are off to tea, by grabbing the bananas, but leaving the box in its original place. Specify the goal in the STRIPS language.

   d. Using ADL notations, modify the initial state description in part (a) and the actions in part (b) to allow for the possibility that an object is too heavy for push, and at such cases the *Push* operator should do nothing.

2. ***Plan space (20%)*** (Textbook 11.10) We constrasted forward and backward state-space search planners with partial-order planners, saying that the latter is a plan-space searcher. Explain how forward and backward state-space search can also be considered plan-space searchers, and say what the plan refinement operators are.

3. ***Planning graphs (30%)*** In the example in the textbook about the "Have a cake and eat it too" problem, we see that the planning graph indicates that a solution is possible at stage 2 (i.e., all goals are present, with none of them conflicting), when there is actually a 2-steps action sequence that achieves that goal. Let's consider only serial planning graph, in which all actions other than persistence actions are automatically mutex to each other.

   a. Suppose the planning problem consists of only two predicates (e.g., Eaten(Cake) and Have(Cake) in the example). Show that it is always the case that *if there is a plan of at most k steps, then the k-th stage will indicate that a solution is possible; and vice versa.*

   b. Give a planning problem consisting of three predicate in which the assertion of part (a) does not hold, and show the planning graph at the level at which the assertion is violated.

4. ***Hierarchical Planning (25%)*** (Textbook 12.4) Give an example in the house-building domain of two abstract subplans that cannot be merged into a consistent plan without sharing steps. (*Hint:* Places where two physical parts of the house come together are also places where two subplans tend to interact.)