Assignment 10

Reinforcement Learning

Prof. B. Ravindran

1. Suppose that in solving a problem, we make use of state abstraction in identifying solutions to some of the sub-problems. In this approach, is it possible to obtain a recursively optimal solution to the original problem?

   (a) no

   (b) yes

2. What kind of solution would you expect to obtain if, in solving a problem, policies for each individual sub-problem are learned in isolation (i.e., without taking into consideration the overall problem)?

   (a) hierarchically optimal solution

   (b) recursively optimal solution

   (c) flat optimal solution

3. Do the policies of individual options need to be defined over the entire state space of the MDP (of the original problem)?

   (a) no

   (b) yes

4. Consider the two room example discussed in the lectures. Suppose you define two options, O_1, to take the agent in room 1 (the left room) to room 2, and O_2, to take the agent in room 2 to the goal state. Assuming that you have appropriately specified the initiation sets and termination conditions for both the options and are trying to learn the individual option policies, would you need to use SMDP Q-learning or would conventional Q-learning suffice?

   (a) SMDP Q-learning

   (b) conventional Q-learning

5. Consider a Markov policy over options $\mu : S \times O \rightarrow [0, 1]$, where S is the set of states and O is the set of options. Assume that all options are Markov. While the policy μ selects options, by considering the primitive actions being selected in those options, we can determine another policy, π, which corresponds to μ, but is a conventional policy over actions. In general, will π also be a Markov policy?

(a) no

(b) yes

6. Consider the following problem design. You have a grid world with several rooms, as discussed in the lectures, with the goal state in a corner cell of one of the rooms. You set up an agent with options for exiting each of the rooms into the other. You also allow the agent to pick from the four primitive actions. There is a step reward of -1. The learning algorithm used is SMDP Q-learning, with normal Q-learning updates for the primitive actions. You expect the agent to learn faster due to the presence of the options, but discover that it is not the case. Can you explain what might have caused this?

   (a) the options are not useful for solving the problem, hence the slowdown

   (b) initially, the agent will focus more on using primitive actions, causing slowdown

   (c) due to the presence of options, we can no longer achieve the optimal solution, hence it takes longer

   (d) it takes longer because we are using SMDP Q-learning compared to conventional Q-learning which is faster

7. Using intra-option learning techniques, we can learn about options even without ever executing them. True or false?

   (a) false

   (b) true

8. Suppose that you have identified a set of sub-tasks for solving a large problem using the hierarchical learning approach. To solve each sub-task efficiently, you want to constrain the primitive actions that can be executed within each sub-task. Specifying such constraints is possible in

   (a) options

   (b) HAMs

   (c) both