Assignment 6 Report for Part A

Kazi Sabrina Sonnet

1a. How many iterations of VI are required to turn 1/3 of the state's green? (i.e., get their expected utility values to 100).

Answer: 4

1b. How many iterations of VI are required to get all the states, including the start state, to 100?

Answer: 8

1c. From the Value Iteration menu, select "Show Policy from VI". (The policy at each state is indicated by the outgoing red arrowhead. If the suggested action is illegal, there could still be a legal state transition due to noise, but the action could also result in no change of state.) Describe this policy. Is it a good policy? Explain.

Answer: Except the goal state, policy in every state point right, where in the goal state it points down. It is not a good policy because the actions in this case are illegal. In a deterministic world like this the goal state can not be reached using this policy. For example, after moving the smallest disc from peg 1 to peg 3, there is a chance the state will not change.

2a. How many iterations are required for the start state to receive a nonzero value?

Answer: 8

2c. At this point, view the policy from VI as before. Is it a good policy? Explain.

Answer: The policy is good because it doesn't provide any illegal action and gives an optimal path from start to goal.

2d. Run additional VI steps to find out how many iterations are required for VI to converge. How many is it?

Answer: 56

2e. After convergence, examine the computed best policy once again. Has it changed? If so, how? If not, why not? Explain.

Answer: No, the policy did not change. Because, the policy directs the states with the highest current value and order accordingly. Further iteration doesn't change this order because states closer to goal state are more likely to converge due to noise propagation.

3a. Run Value Iteration until convergence. What does the policy indicate? What value does the start state have? (start state value should be 0.82)

Answer: The start state has the value 0.82. This policy never directs towards the 3rd peg's two discs which is silver goal. It remains closest to the start state because for the high reward (10)

3b. Reset the values to 0, change the discount to 0.9 and rerun Value Iteration until convergence. What does the policy indicate now? What value does the start state have? (start state value should be 36.9)

Answer: Start state has the value 36.9. The policy directs from the start state to the gold goal. Because it wants to grab a big reward than a discounted reward.

4a. In how many of these simulation runs did the agent ever go off the plan?

Answer: 8

4b. In how many of these simulation runs did the agent arrive in the goals state (at the end of the golden path)?

Answer: 7

4c. For each run in which the agent did not make it to the goal in 10 steps, how many steps away from the goal was it?

Answer: 1 step

4d. Are there parts of the state space that seemed never to be visited by the agent? If so, where (roughly)?

Answer: Roughly, the big top triangle and the small two triangles below the top one.

5a. Since it is having a good policy that is most important to the agent, is it essential that the values of the states have converged?

Answer: It is not necessary that the values of the states have converged if the policy remains same.

5b. If the agent were to have to learn the values of states by exploring the space, rather than computing with the Value Iteration algorithm, and if getting accurate values requires re-visiting states a lot, how important would it be that all states be visited a lot?

Answer: There are some states those are slower than other states in terms of converging. The agent doesn't need to visit all those states but only the slower converging states.

Option B:

I implemented epsilon greedy Q learning including custom epsilon, fixed epsilon, custom alpha and fixed alpha, exploration function, user driven Q learning framework. At the beginning of the exploration there is high alpha and high epsilon value but at the end of reinforced exploitation the alpha and epsilon value is low. For the fixed alpha and epsilon, it took a long transition where the custom alpha and epsilon function the exploration of the Q values is much faster

The performance seems much better when I used exploration function than fixed alpha and epsilon. In addition, exploration function with the custom epsilon certainly increases performance. Exploration function can ensure better exploration in overall state.