COMP3702 Artificial Intelligence (Semester 2, 2022)

Assignment 3: HexBot Reinforcement Learning

Name: Eskil Pedersen

Student ID: 47613722

Student email: e.pedersen@uqconnect.edu.au

Note: Please edit the name, student ID number and student email to reflect your identity and **do not modify the design or the layout in the assignment template**, including changing the paging.

**Question 1** (Complete your full answer to Question 1 on the remainder of page 1)

Q-learning is closely related to the Value Iteration algorithm for Markov decision processes.

**a)** (5 marks) Describe two key similarities between Q-learning and Value Iteration.

Both algorithms are used to solve a version of a Markov Decision Problem (MDP). This is a description of an environment used in reinforcement learning, which is the first key similarity: Both value iteration and Q-learning are reinforcement learning algorithms. Said in another way, when Q-learning and Value Iteration start, they don’t know which action is best from each state, but through trial and error, they try to learn what works best and what doesn’t.

A second key similarity builds on the fact that both algorithms try to maximize the cumulative expected reward. The rewards are stored in a table containing values for all explored states, which are updated as the algorithms run. The computed values in both Value Iteration and Q-learning get better over time and are guaranteed to converge, given enough computational power. When the values have converged the agent knows the optimal action from every state.

**b)** (5 marks) Give one key difference between Q-learning and Value Iteration.

One key difference between Value Iteration and Q-learning is that Value Iteration is model-based, and Q-learning is model-free. A model in reinforcement learning is a function that maps a pair of states and actions to probabilities of ending up in different states. Since Value Iteration is model-based, it knows the transition function, however Q-learning doesn’t know these probabilities. Since Q-learning doesn’t know the probabilities, the agent needs to exploration to learn the probabilities and rewards.

**Question 2** (Complete your full answer to Question 2 on page 2)

**a)** (5 marks) Explain the difference between off-policy and on-policy reinforcement learning algorithms. Explain where this difference is represented in the Q-learning and SARSA algorithms.

An off-policy reinforcement algorithm learns the value of all policies, no matter what this policy is. On-policy reinforcement learning learns the value of the policy that has the best value at that moment. The off-policy algorithms try to learn the value through exploration, even though the policy being explored sends the agent to an unpreferable state.

In Q-learning, when updating the Q-table, the next state is chosen by an exploration strategy. However, in SARSA after an action is selected with an exploration strategy, this action is saved and will be used in the next exploration phase. When the next episode is started, the exploration strategy will again be used to find the action.

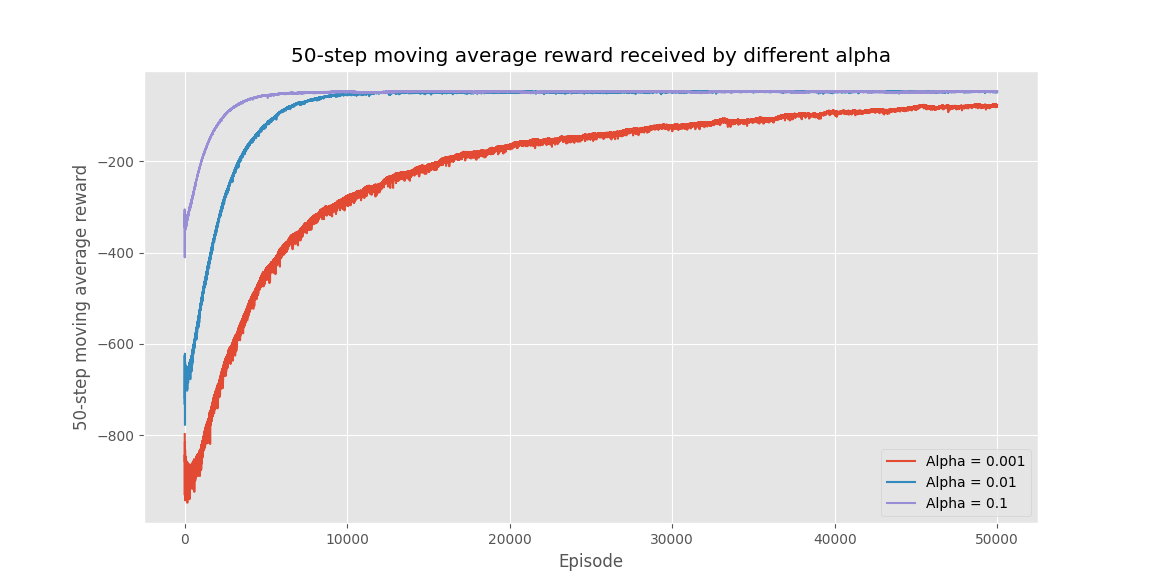
**b)** (5 marks) How does the difference between off-policy and on-policy algorithms affect the way in which Q-learning and SARSA solve test cases ex3.txt and ex4.txt? If you were unable to solve these test cases, it is sufficient to answer with reference to what you think would happen, based on the setup described in the test case files.

Both Q-learning and SARSA choose the shorter, and riskier path close to the hazards. I think it makes sense that Q-learning chooses the riskier path because Q-learning explores all actions, no matter the previous value stored for this action. However, I would think that SARSA had chosen the less risky, but the long way around the obstacles. This is because SARSA is on-policy learning, and I would think that the first action from the initial state would be to go the less risky path around the obstacles because this probably has a better short-term reward.

**Question 3** (Complete your full answer to Question 3 on page 3)

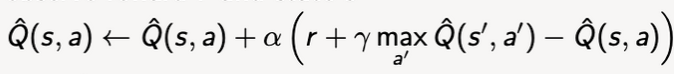
**a)**

(5 marks) Plot (all on a single plot) the quality of the policy learned by Q-learning in test case ex3.txt against episode number for three different fixed values of the learning\_rate (which is called α in the lecture notes and in many texts and online tutorials), as given by the 50-step moving average reward (i.e. for this question, do not adjust α over time, rather keep it the same value throughout the learning process). Your plot should display the solution quality up to an episode count where the performance stabilizes. This may take a significant number of episodes (e.g. >50,000) depending on the learning rates used. Note that the policy quality may still be noisy, but the algorithm’s performance will stop increasing and its average quality will level out. Your plot should include axis labels and a legend.

****

**b)** (5 marks) Discuss the effect of varying the learning\_rate. You should make reference to the Q- learning algorithm to support your discussion. If you were able to generate a plot in Q3a, you may also make reference to this in your discussion.

The difference in learning rate can be seen in the three different plots in Q3a. The Q-learning algorithm updates the Q-values by some learning factor-alpha. If this learning factor is high, the Q-values will be updated faster by the new values seen in the parenthesis. When alpha is low, the update should be much slower. As you can see from the three plots when the learning rate is 0.1 the episode rewards converge after around 5000 episodes. However, when the learning rate is 0.01 and 0.001 respectively, the episode rewards converge after 9000 and 60 000 episodes.



**Question 4** (Complete your full answer to Question 4 on page 4)

**a)**

(5 marks) Plot (on a single plot) the quality of the learned policy against episode number under Q- learning and SARSA in test cases ex3.txt and ex4.txt respectively, as given by the 50-step moving average reward. Your plot should display the solution quality up to an episode count where the performance of both algorithms stabilizes. Your plot should include axis labels and a legend.

Chart

Description automatically generated with medium confidence

**b)**

(5 marks) With reference to your plot, compare the learning trajectory of the two algorithms, and their final solution quality. Discuss the way the solution quality of Q-learning and SARSA change as they learn to solve the testcase, both as they learn and once, they have stabilized.

As we can see, in the first 3000 episodes approximately SARSA has a better reward than Q-learning. This is expected since SARSA learns the values of the current policy that has the best value. However, after around 3000 episodes, Q-learning starts to get the same rewards as SARSA. This is due to the fact SARSA keeps learning the value of the best policy, while Q-learning also learns the value of other policies. This is decided by either UCB or epsilon greedy. In the plot over, epsilon greedy is used. From around 6000 episodes can be seen that Q-learning has passed SARSA in average reward, and from around 8000 episodes, they have both stabilized. Q-learning continues to have a slightly better reward than SARSA. This can be explained by the same reason why Q-learning after around 3000 episodes almost has the same reward as SARSA.