COMP3702 Artificial Intelligence (Semester 2, 2022)
Assignment 3: HEXBOT Reinforcement Learning

Name: Onai Chenje

Student ID: 45311994

Student email: o.chenje@uq.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)

Key similarities between Q-Learning and Value Iteration:

1. Both estimate cost/value of performing an action a on state s, storing and updating them until they converge by performing actions on each state.
2. Updates stored values Q(s,a) which store the estimated reward/cost of taking an action a on a state s, in order to converge to an optimal policy where reward is maximised.

Key difference between Q-Learning and Value Iteration:

1. In VI, the transition and reward functions are given or can be found, and then directly applied to get Q(s, a). In Q-Learning, as these functions are not provided, Q(s, a) needs to obtain an estimated discounted cost/value by evaluating observed data i.e., learning

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

a) Difference between off-policy and on-policy reinforcement learning algorithms

- **On-policy algorithms**: Learns the value of the policy being followed. In SARSA, updates to Q-value are done via estimates following a policy, which is updated along with the values. Thus, for Q(s', a') a' is provided by following the policy on s'.

- **Off-policy algorithms:** Learns the value of the optimal policy regardless of what actions it performs on state S. In Q-Learning, the estimate is provided via a greedy policy, which selects the next action a' to perform on the next state s' based on its ability to maximise Q(s', a'). It does not consider the true nature of the policy.

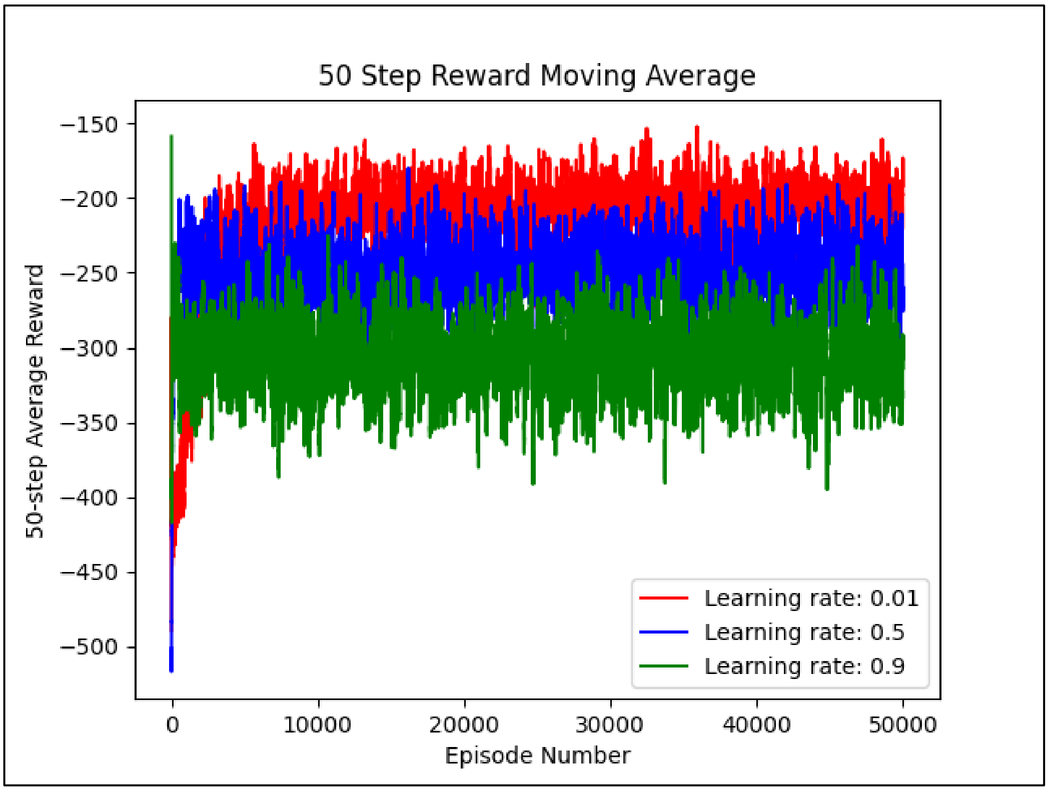
b) Off-policy vs on-policy for tests 3 and 4

Via off-policy algorithms, the Hexbot in test 3 follows a greedier policy aimed at maximising the value to be obtained by a next state. As such, it was much more likely to take riskier paths during training, which in turn yielded it a higher training reward.

Oppositely, Via on-policy algorithms the Hexbot in test 4 was more risk adverse, as it was following and updating the followed policy. Thus, it yielded a lower training reward than test 3. Although both bots took the same path, Q-Learning was following a greedier policy, whereas SARSA found it to be the optimal path to take in test 4.

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

a) Plot of the quality of the learned policy by learning rate



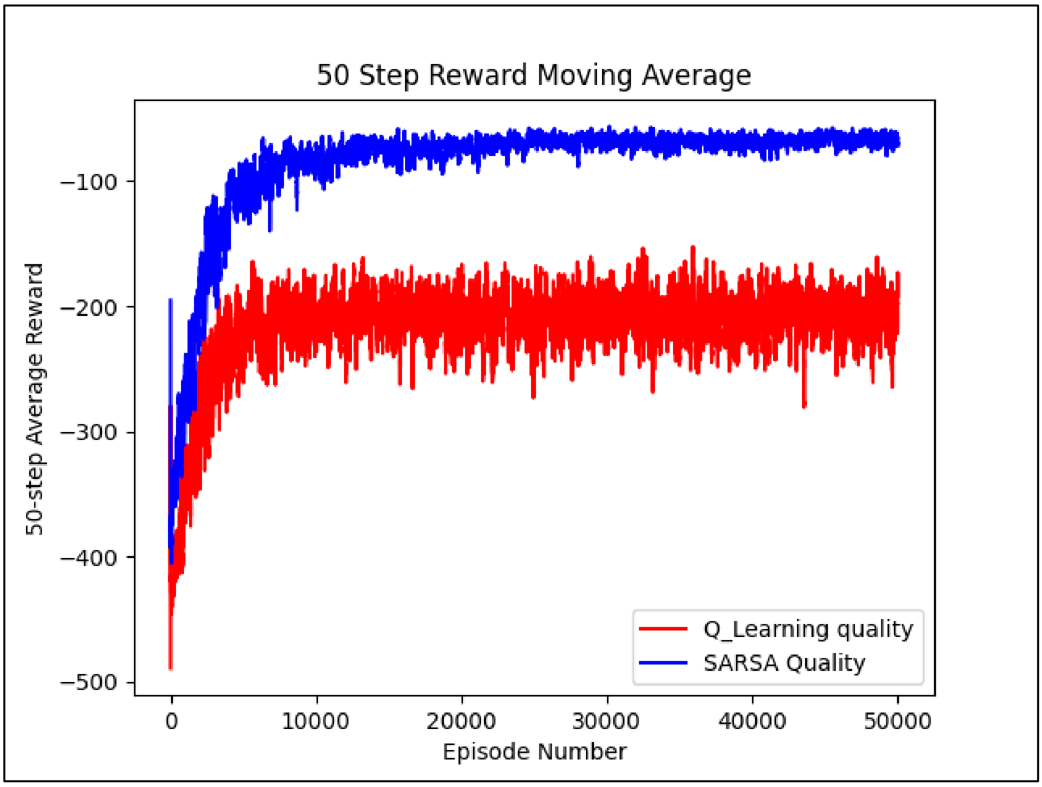
b)

Varying the learning rate will influence the rate at which learning occurs by influencing each data points affect on the estimated value. This is to say, a higher learning rate means a higher influence and lower rate means a lower influence. In the formula, this influences the effect of the estimate provided by s, a, s' and a' $\alpha(r + \gamma \max_{a'} Q(s', a') - Q(s, a))$, with 0 preventing an update and 1 providing a direct update of this estimate into Q(s, a).

On the plots above, the learning rate influences the effect of new estimates per episode on the average reward. In the green plot, the high learning rate indicates that the provided estimates will have a larger effect on the Q-values. If a value is less useful or less accurate (i.e., a much lower cost), it can also have a large effect on the Q-value. The red plot on the other hand has a lower learning rate, and thus the estimates it provides during an episode have less of an effect on the Q-table; this is why the graph presented above yields the highest average reward.

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

a) QL vs SARSA for tests 3 and 4



b) Comparison

Based on the trajectories of the plots above, SARSA has a higher solution final quality for this problem. Both tests have a similar graphical appearance but as number of episodes grows, however as Q-Learning follows its greedier policy, it stabilises at a lower average reward as the episodes it traverses are more susceptible to higher costs/lower rewards accrued from following a more dangerous policy.

SARSA took a less greedy approach and followed a policy that was optimised through iteration. Thus, it took less risks, and was not as affected by the same negative high cost/low reward paths that the Q-learning solution followed.

-