Chase Zhou

ECE 448

9/26/14

MP1 Part 2

1.

a.) The Q-learning agent gets a negative total reward on the first episode and gets 0 total reward for all the following episodes. From the policy simulator, it looks like the robot does not move.

b.) The Q-learning agent starts off with negative total rewards in the first few episodes, but gets better until it ends up with a total reward in the hundreds in the final few episodes. The Q-learning agent does this because of the chance of making a random move. Without the epsilon factor, the agent settled for a local maximum of 0 total reward. Once it had a chance to move randomly, it overcame that and was able to find more rewards.

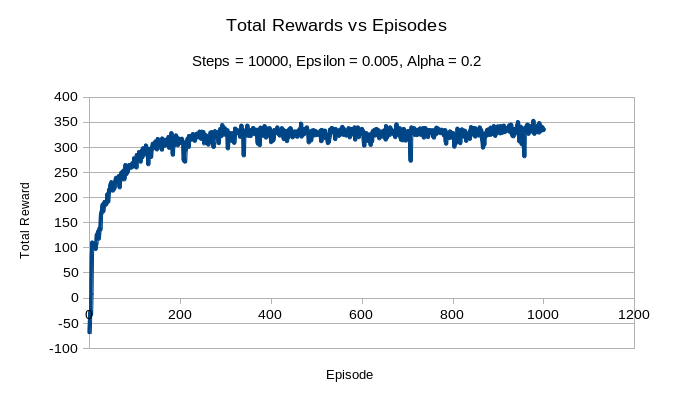
c.) With epsilon = 0.5, the Q-learning maintains a negative total reward for most of the episodes. This is because it has a 50% chance of making a random move. Therefore, it only behaves optimally 50% of the time. It then doesn't really make decisions based on maximizing its reward for most of the time. This is why its total rewards does not increase.

2.

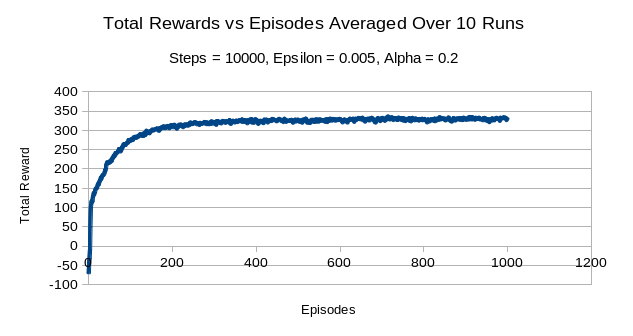
a.) The agent improves slightly over the episodes but continues to have a negative total reward throughout all episodes. This is because the agent is not aware of the thief that causes it to incur heavy penalties. It continues to lose reward by continuing to meet the thief because it cannot avoid him.

b.) The agent quickly improves through the episodes. It is able to consistently reach total rewards in the high 200's by the end of the 1000 episodes. This is because the agent is now aware of the thief and is able to avoid him. Then, it is able to deliver the packages and return without incurring heavy penalties and so is able to build up its total reward.

c.) After experimentation, it seems the best values are epsilon = 0.005 and learning rate = 0.2. I first tried multiple values of epsilon. I decreased it until I found the value that looked like maximized total rewards. Then I increase the learning rate until I found the value that maximized total reward. These were the values I arrived at and they ended up consistently giving a total reward in the mid 300's. I then tried decreasing and increasing the values again, but none of them had a better result than the values I arrived at before.

3.

The agent's improvement resembles a logarithmic graph. It's initial improvement is very high, but it starts to tapers off at around 200 episodes and improves only slightly until 1000. There are also several dips where total reward drops significantly for one episode.



The overall shape of the graph for the average of 10 simulations is very similar to that of the one simulation. However, it does not have significant dips in total reward. This is because those dips were the result of unfortunate choices made form the epsilon factor and so happened randomly. When averaged across 10 runs, those dips disappear because on average, the value at those points should be around the expected amount. The graph is then much smoother.