CS 440 MP 1 Part 2

Shunhua Fu

1. In **WorldWithoutThief**
    a. With ε set to 0.0, the agent does not perform random actions except perhaps for ties. Besides the first episode, all the other 999 episodes have final rewards of 0.0. This must be because the first episode has a negative total reward and there is no randomness in selecting an action, so the "best" action is always just one that leads to a total reward of 0.0.
    Mean = -0.0065 (this varies only based on the total reward of the first episode)
    Median = 0.0
    Mode = 0.0
    b. Keeping all other settings constant, increasing ε makes the learning algorithm worse and worse. By this, I mean that the actions taken are not as optimized for the highest total reward as ε increases. This is because a higher ε means taking random actions more often. As you will see, as ε increases, the mean, median, and mode will add drastically decrease.
    ε = 0.1
    Mean = 136.79
    Median = 144.5
    Mode = 165.5
    ε = 0.2
    Mean = 97.92
    Median = 102
    Mode = 104
    ε = 0.3
    Mean = 55.57
    Median = 55.75
    Mode = 39.5
    ε = 0.5
    Mean = -28.84
    Median = -28.5
    Mode = -31
    ε = 0.8
    Mean = -82.33
    Median = -83.25
    Mode = -83
2. In **WorldWithThief**
    a. **ε = 0.05**
    knows_thief = n
    Mean = -12.91

Median = -13

Mode = -12

The performance of my agent is pretty mediocre. Since it does not know where the thief is, it can only take actions based on where the thief was in the past.

b. **ε = 0.05**

knows_thief = y

Mean = 274.82

Median = 280

Mode = 286.5

The performance of the agent is significantly better once it is sensitive to the location of the thief. This is because meeting with the thief causes the robot to lose all of its packages, thereby incurring a negative reward, which is bad so the algorithm adjusts to avoid this scenario.

c. Based on the experimentation in 1b, it appears that increasing ε lowers the agent's effectiveness. It then follows that the best ε is very close to 0, but it can't be 0 as shown by the results in 1a. I then played around with the learning rate while keeping everything else constant (ε = 0.05 and knows_thief = y).

rate = 0.05

Mean = 265.37

Median = 273.5

Mode = 274

rate = 0.1

Mean = 274.61

Median = 278.5

Mode = 284

rate = 0.25

Mean = 266.66

Median = 270

Mode = 278.5

rate = 0.5

Mean = 238.64

Median = 241

Mode = 243.5

rate = 0.75

Mean = 198.73

Median = 202

Mode = 204

The general trend of varying the learning rate seems to indicate that the optimal learning rate is somewhere around 0.1. Testing rate = 0.075 and rate = 0.125 gave results lower than rate = 0.1. So the peak result should be found at or just slightly greater than rate = 0.1.

ε = 0.01

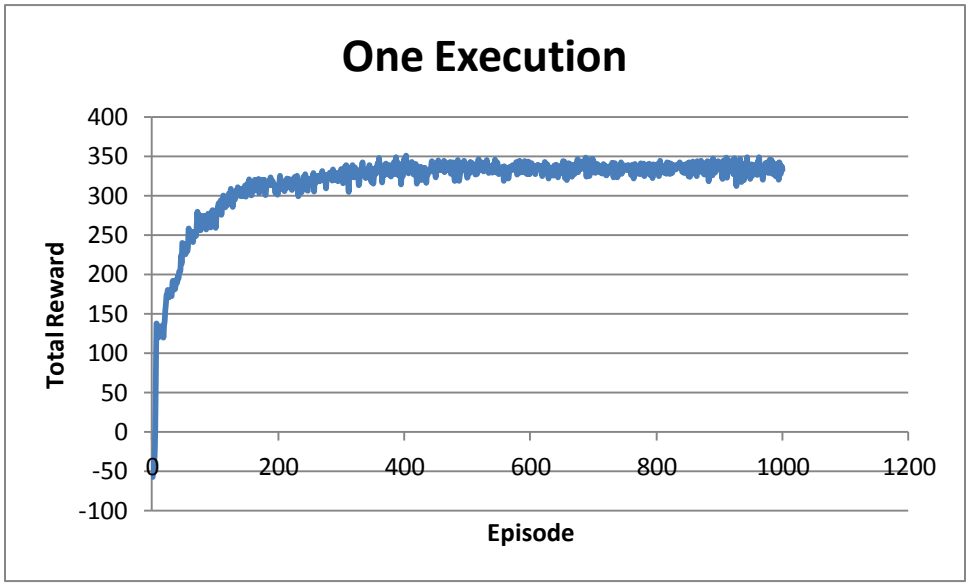
rate = 0.11

Mean = 319.45

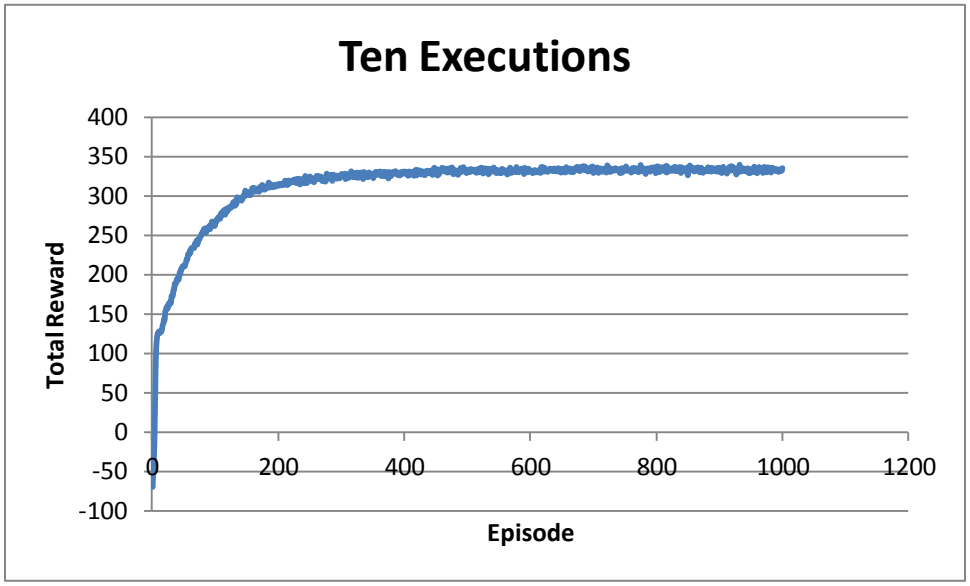
Median = 332.5

Mode = 335

3.



You can clearly see that the agent is learning over the first 200 or so episodes so that it reaches a pretty optimal amount total reward by episode 350 or so.



The variation in the data narrows in the average of ten executions of the simulation. This means that the agent is reaching a similar optimal total reward in each execution.