Jaime P Roque
roque2
9/29/2014

CS440 – MP1-2

1) WorldWithoutThief

a. With a discount factor of 0.9, a learning rate of 0.1, and ε of 0.0, the agent briefly accumulated negative rewards before plateauing at a net value of zero rewards for the cast majority of its running time. In our ε-greedy algorithm, ε denotes the chance that the agent will choose a random action such that there is a possibility of discovering new rewards in the case it finds a set route to accumulate rewards. However, in this case ε was set to 0, meaning that the agent did have any chance of discovering new rewards and instead traversed across space where it knew the rewards. This caused the agent to plateau at a net value of zero for each episode of 10000 steps.

b. When ε is set to 0.1, the agent begins accumulating a net positive amount of rewards for the latter parts of its run. As explained previously, ε indicates the chance that the agent will choose a random action instead of intentionally choosing an action based off of know state-action-qValue tuples. As ε is no longer 0, the agent now has a chance of deviating from its known reward track, with said reward track having potential of locking the agent at 0 rewards, and discovering new spaces with rewards of greater value.

c. When ε was set to 0.5, the agent started to accumulate a combination of net negative and net positive rewards across each of its episodes. Again, ε indicates the chance that the agent will deviate from its known path and choose a random action, with the potential of discovering greater rewards. However, with a ε value of 0.5, the agent has half a chance of randomly choosing an action and half a chance of following the known reward policy. As such, the agent will start to behave erratically when compared to the behavior of an agent with a small ε with a positive value, and will not necessarily have a positive net reward per episode.

2) WorldWithThief

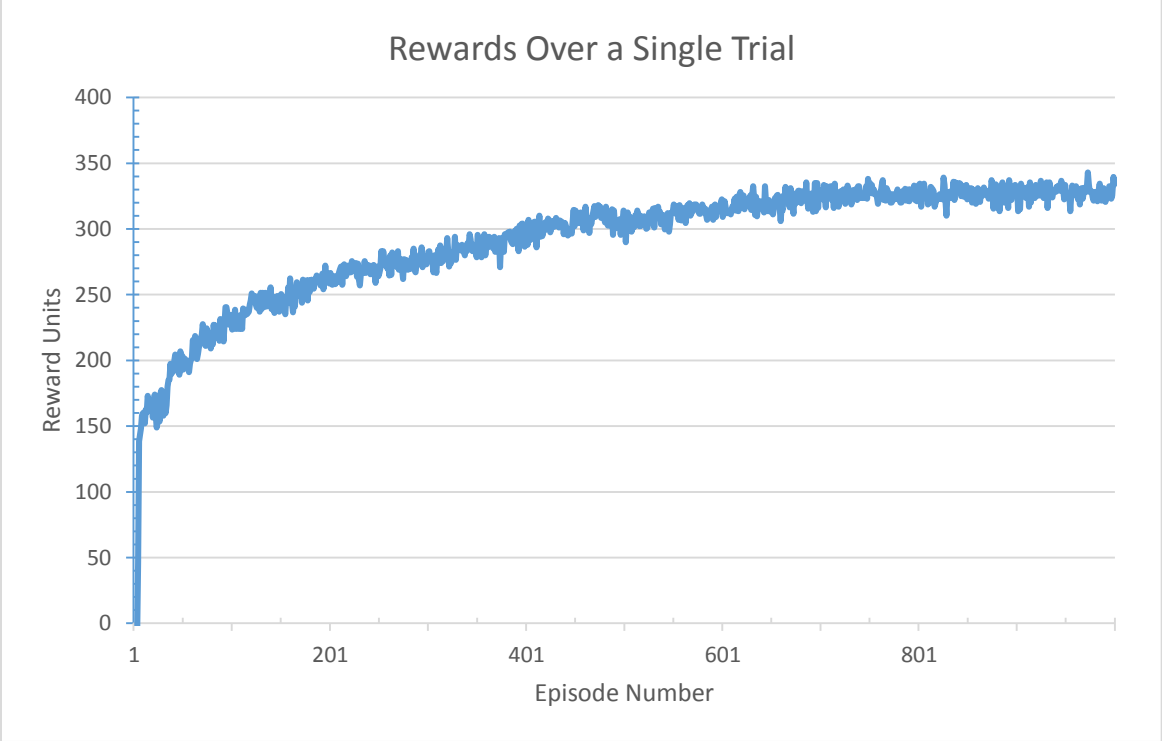
a. With a ε value of 0.05 and the knows_thief parameter set to n, the agent accumulated a negative net value across the vast majority of its episodes. By setting the knows_thief parameter to n, the agent is completely unaware of the thief's presence. This prevents the agent from modeling its behavior around the presence of the thief, significantly increasing its chances of running into it and incurring a penalty, as seen in its episodic run.

b. When the knows_thief parameter is set to y, the agent accumulates overwhelmingly positive net rewards per episode when compared to the episodic run where knows_thief is set to n. Since the agent is now aware of the thief's

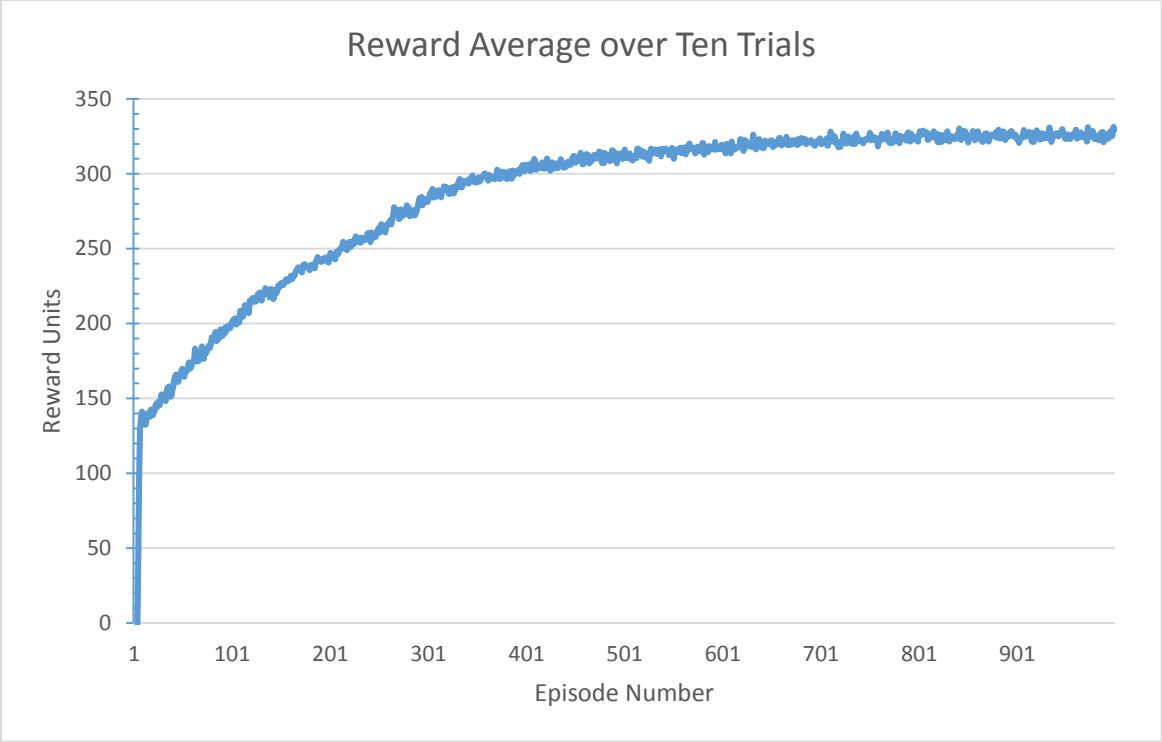
presence and location, it models its policy such that it will avoid the thief as much as possible and avoid any penalties.

c. In this investigation, I considered six general situations: the situation in which the learning rate was greater than ε, the situation where ε was greater than the learning rate, situations where the learning rate was high or low, and situations where ε was large or small. For the first two situations, I found that resulting rewards are larger if the learning rate is greater than ε. Next, larger learning rates increase the rewards gained, but values greater than 0.05 gradually reduce the reward gains the larger the rate is. For ε, its value must be positive to give the possibility of discovering new rewards but it must also not be too large to prevent the agent from foregoing already known rewards. After experimenting with various values, I found that a learning rate of 0.05 and ε of 0.01 give the best values.

3) Data



As we can see in the graph above, over a single trial series of 1000 episodes of 10000 steps, the net rewards gained per episode behaves logarithmically, levelling off at roughly 325 reward units per episode.



In the graph above, we do the same thing previously except we average the results of ten trials to get a more consistent graph. Again, we see that the net rewards per episode increases logarithmically, levelling off at roughly 235 reward units per episode. This is consistent with the expected behavior of the agent as there is only so much the agent can learn in a mostly static world, with the position of the thief being the only changing part. Therefore, after a certain number of episodes the number of reward units that the agent gains per episode eventually levels off, with variations occurring during the few instances in which the agent loses the packages due to either slipping or encountering the thief.