1. In **WorldWithoutThief**

(a) When ϵ =0, all the rewards for the episodes are 0.

This is because firstly, the agent does not perform random actions when ϵ =0 for the randomly generated number can never below a zero ϵ.

Moreover, there is no policy for the Q-learning agent to comply with, it cannot take actions according to a certain fixed sequence of actions.

Finally, Q-learning agent will take action that can bring maximum Qvalue for current state, but after initialized at the very beginning of the simulation, all the Qvalues are zero thus the agent cannot make a first move.

(b)When ϵ =0.1, rewards of first 26 episodes are negative or small positive value. Then rewards boost and oscillate around 150 and the range of rewards is huge, from near 10 to 200.

Firstly, rewards are around 0 at the beginning because the agent is learning about the world and it is common to make mistakes (slip).

Secondly, rewards increase after some episodes because the world is kind of learned and matrix of Q value has been built by the agent, now agent can choose actions generating more Q value which means more possible leading to high rewards.

Finally, rewards oscillate because though will the rewards converge to a certain level, the world is stochastic thus choosing actions having high Q value does not guarantee high reward in *a certain* episode. Moreover, ϵ >0 means it is possible that agent choose actions randomly and it may choose actions leading to negative rewards.

(c) When ϵ =0.5, rewards are negative.

Obviously there is much possibility that agent will choose actions randomly because there is much possibility that the generated random number is smaller than a large ϵ. This agent will not benefit as much to the Q value it generated after episodes of exploration thus are kind of blind to the world information.

2.In **WorldWithThief**

(a) When ϵ =0.05 and **knows_thief** is n, rewards are negative and remain a low level.

Because the agent do not know the position of the thief, it cannot learn the information of thief and "predict" the thief's move to avoid the negative reward generated by encounter the thief.

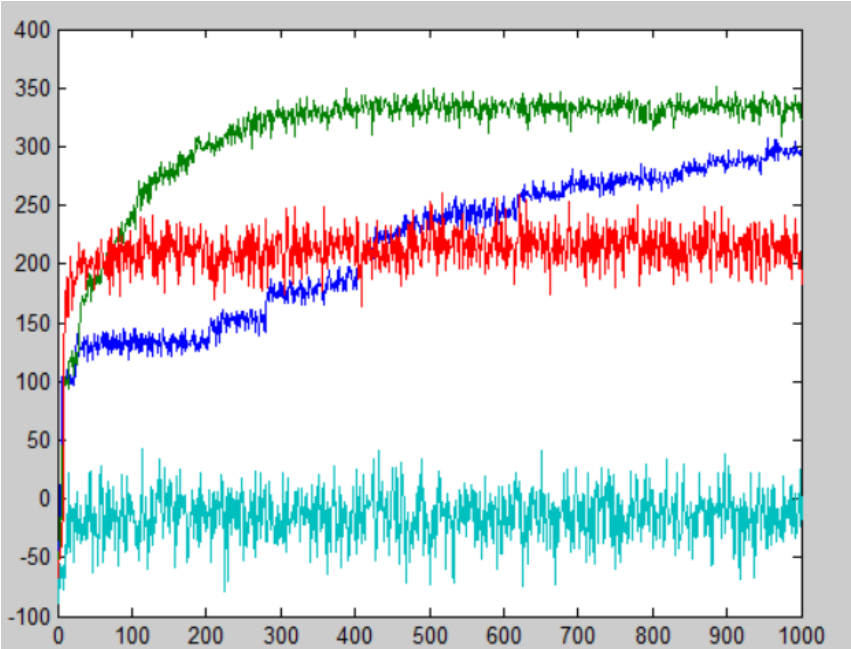
(b)When **knows_thief** is y, rewards increase from negative value and converge to about 270.

Now the agent is aware of position of the thief, thus it can differentiate states when it is in same position but thief is in different position, enlarge the matrix of Q value, and choose action to avoid encountering the thief just like to "predict" the thief's move. And this is why agent can get more rewards than in question a.

For example, taking company as (0,0), consider that agent is in(1,2). If thief is in (2,2), then the action to move east has higher Q value than if thief is in (2,3).

(c)

Figure1: Set learning rate=0.1 and compare outcomes of different ϵ



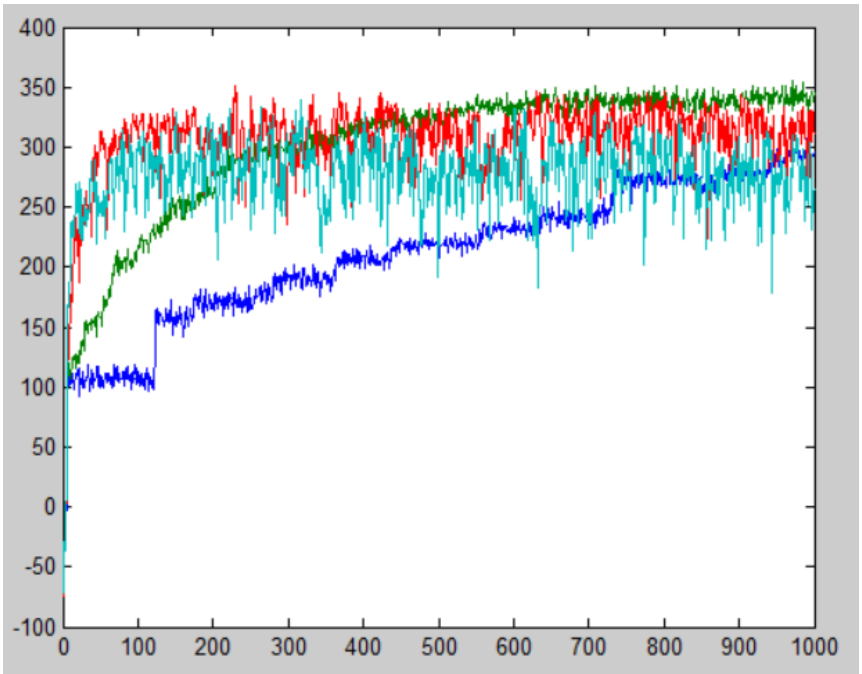
Color of line	ϵ
gray	0.3
red	0.1
green	0.01
blue	0.001

When ϵ is small (0.001), rewards converge to optimal value vary slowly. Agent is kind of conservative.

When ϵ is larger than 0.1, rewards tend not to converge to optimal value and even fall below zero. Agent behaviors are too random.

The best ϵ is between 0.005 and 0.01.

Figure2: Set ϵ =0.01, compare outcomes of different learning rate



Color of line	Learning rate
gray	0.8
red	0.5
green	0.1
blue	0.025

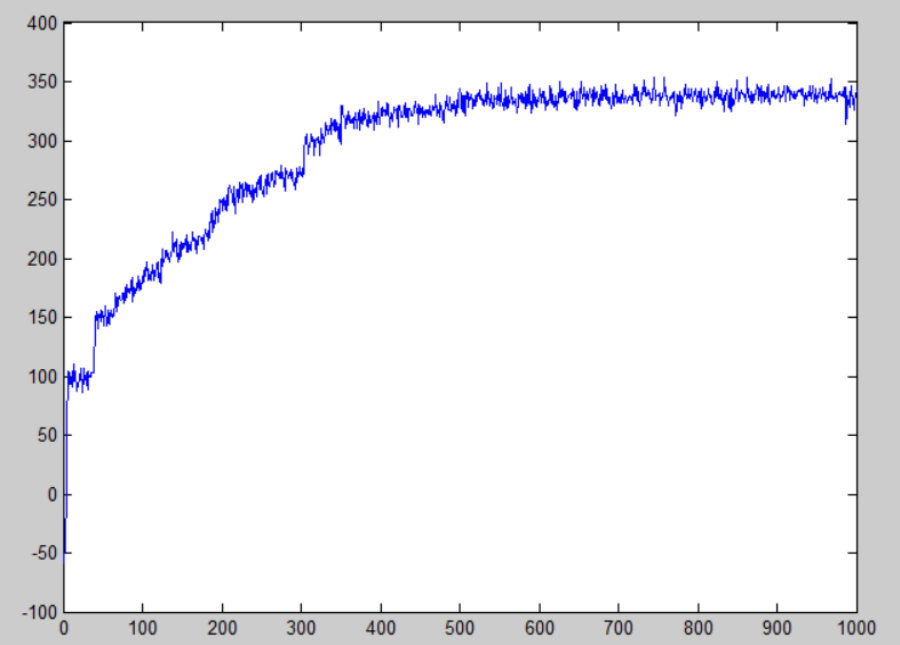
When learning rate is small (0.025), rewards converge to optimal value vary slowly. When it is 0, agent cannot benefit from exploring and keep getting negative rewards.

When learning rate is large, rewards tend to fluctuate and the larger the rate is, the bigger the rage of fluctuation. When it is 1, agent is forgetful and cannot benefit from the experience.

The best learning rate is 0.1.

3. Set learning rate=0.1, ϵ=0.005.

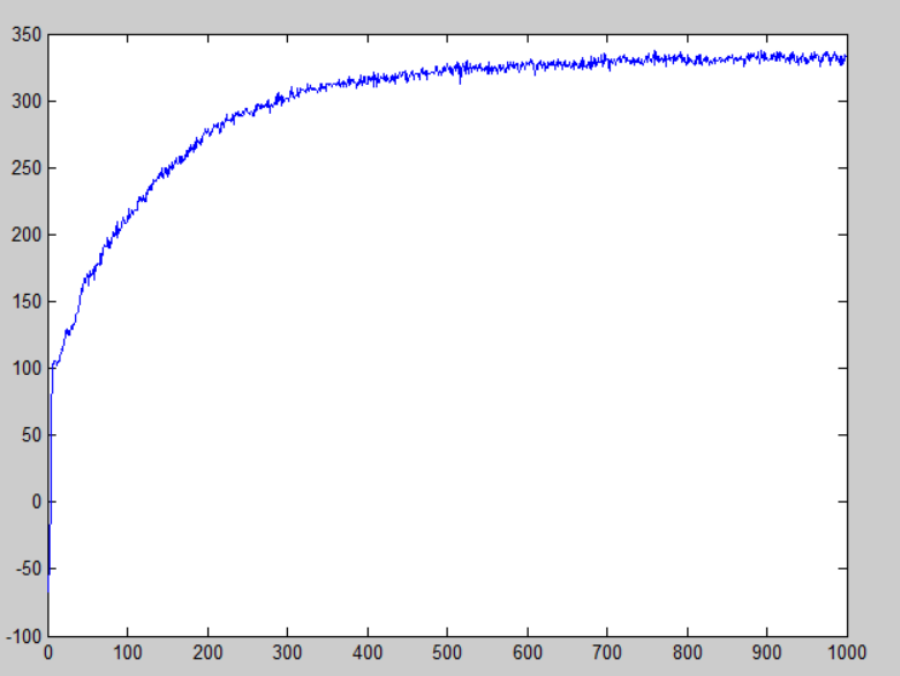
Figure3: Rewards of 1000 episodes of first simulator



The rewards of first several episodes are negative and increase quickly in first 350 episodes. This is a period for the agent to explore the world and acquire knowledge.

Then the rewards converge to a big value (about 340) and fluctuate because of random behaviors and the nature of world: stochastic.

Figure4: Rewards of 1000 episodes averaged over ten simulations



Compared with plot of first simulation, the plot here is smoother and fluctuation is small because averaging partially remove the influence of random action.

Moreover, the fact that the plot converge to same level indicates that an optimal solution is built by the agent.