MLGT HW2 Report

Gyuseok Lee
Institute of Software Research
Carnegie Mellon University
gyuseokl@andrew.cmu.edu

1 Introduction

This homework is designed for understanding the concept of the MADDPG and implementing the codes. Especially, given ranger/poacher scenario, each agent has a different actor-critic structure where actor takes an action to maximize the cumulative reward and critic is used for calculating the reward (Q - function). So, I would like to explain about how to get the result (i.e., 3 learning curves) and show the visualization (i.e., gif files) when using reward shaping or not.

3. Task2: Testing the implementation

2 Task 1: Implementing MADDPG

In this section, I have to implement MADDPG in python program. So, I upload my code into Canvus. Briefly, I implement the q function, q loss, p function, p loss and q targets, which are necessary elements for updating the policy and evaluating the Q-function well.

3 Task 2: Testing the implementation

3.1 Task 2.1

This task is result for 5000 episodes, taking 10-15 seconds per 1000 episodes and receiving the reward -6. So I would like to show how much the agent get the reward through the episodes. As you can see the figure 1, its reward increase more than before.

3.2 Task 2.2

This task is result on ranger-poacher scenario for 3000 episodes. Our trained policies receive rewards of [-0.06, 0.06, 0.06] after training for all 3000 episodes, taking 60-100 seconds per 500 episodes. I attached the plot for each agent's learning curves like below.

3.3 Task 2.3

This task is showing the GIF file and analyze the result. As you can see, it seems not to learn well. For example, even though ranger are close to poacher, ranger cannot catch it. Plus, poacher also didn't catch the animals.

33rd Conference on Neural Information Processing Systems (NeurIPS 2019), Vancouver, Canada.

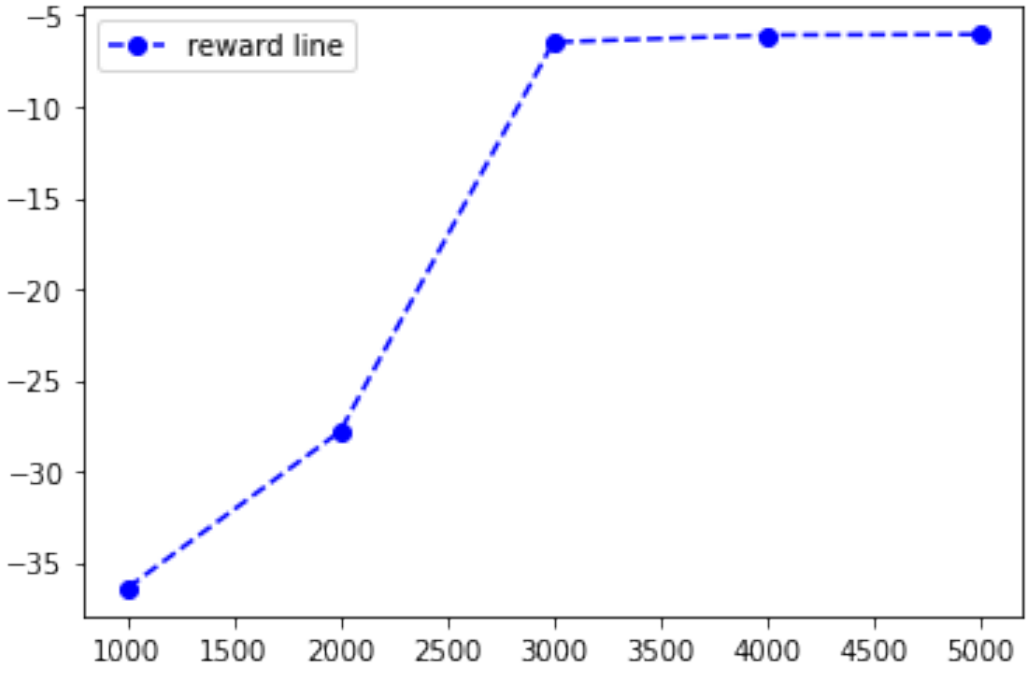

Figure 1: Result for Task 2.1

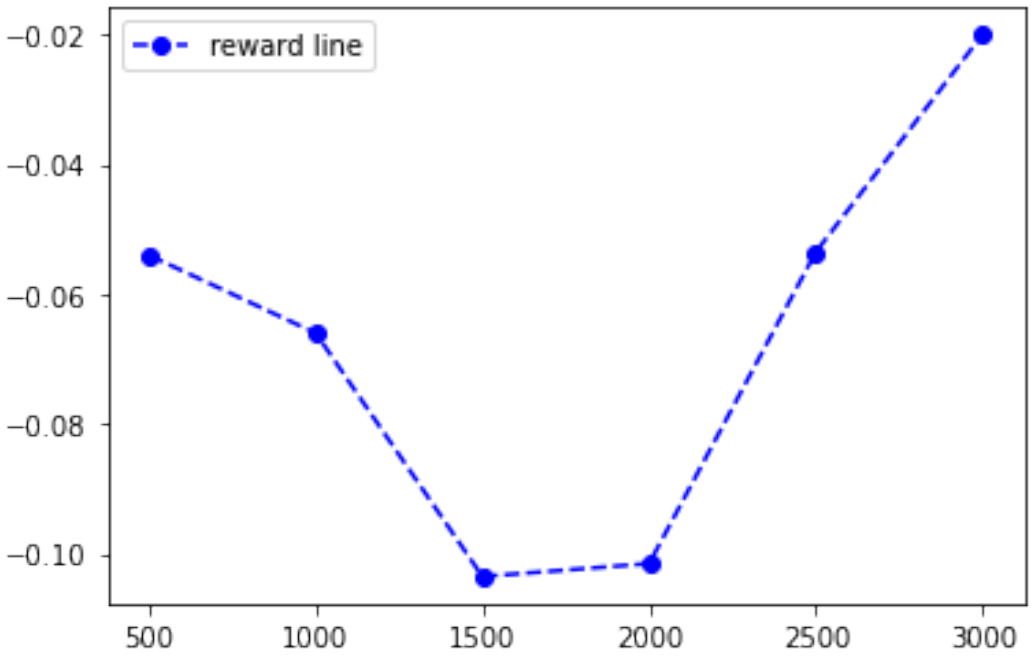

Figure 2: Poacher's reward for Task 2.2

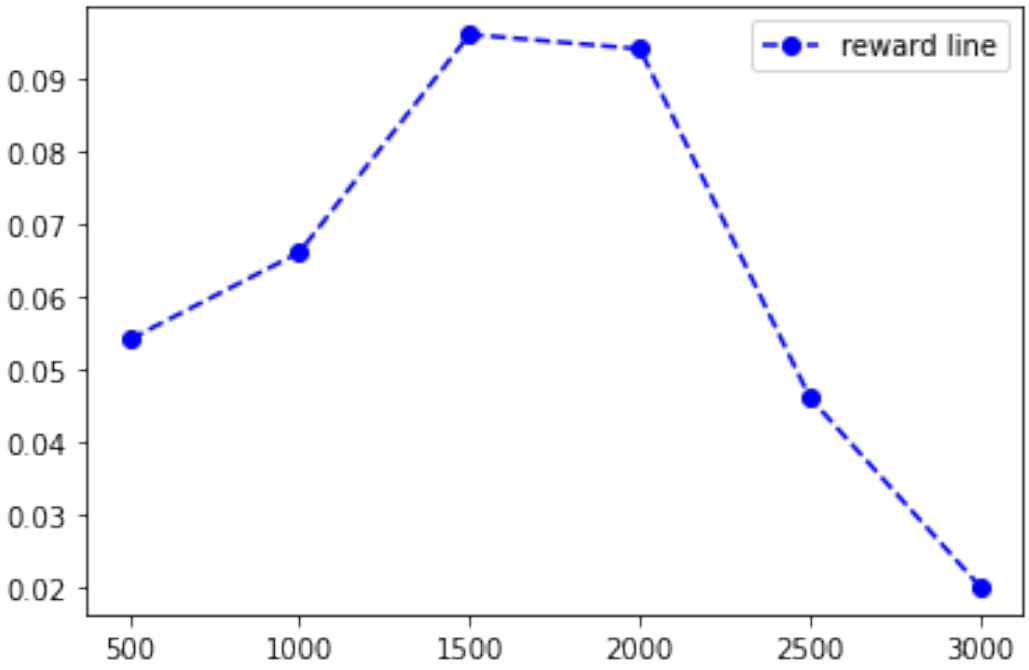

Figure 3: Ranger's reward for Task 2.2

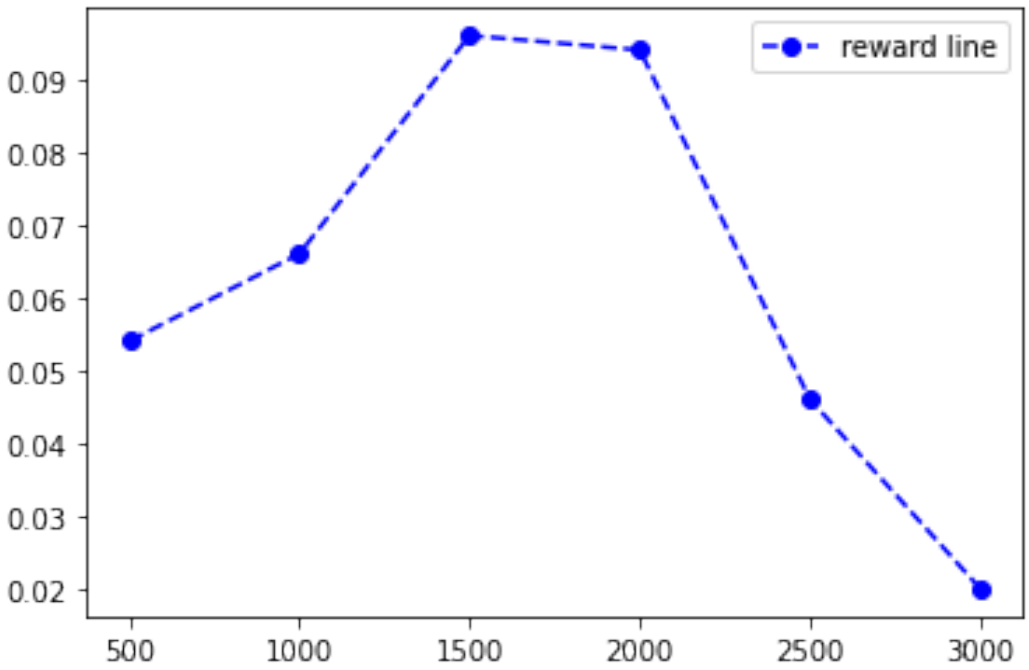

Figure 4: UAV's reward for Task 2.2

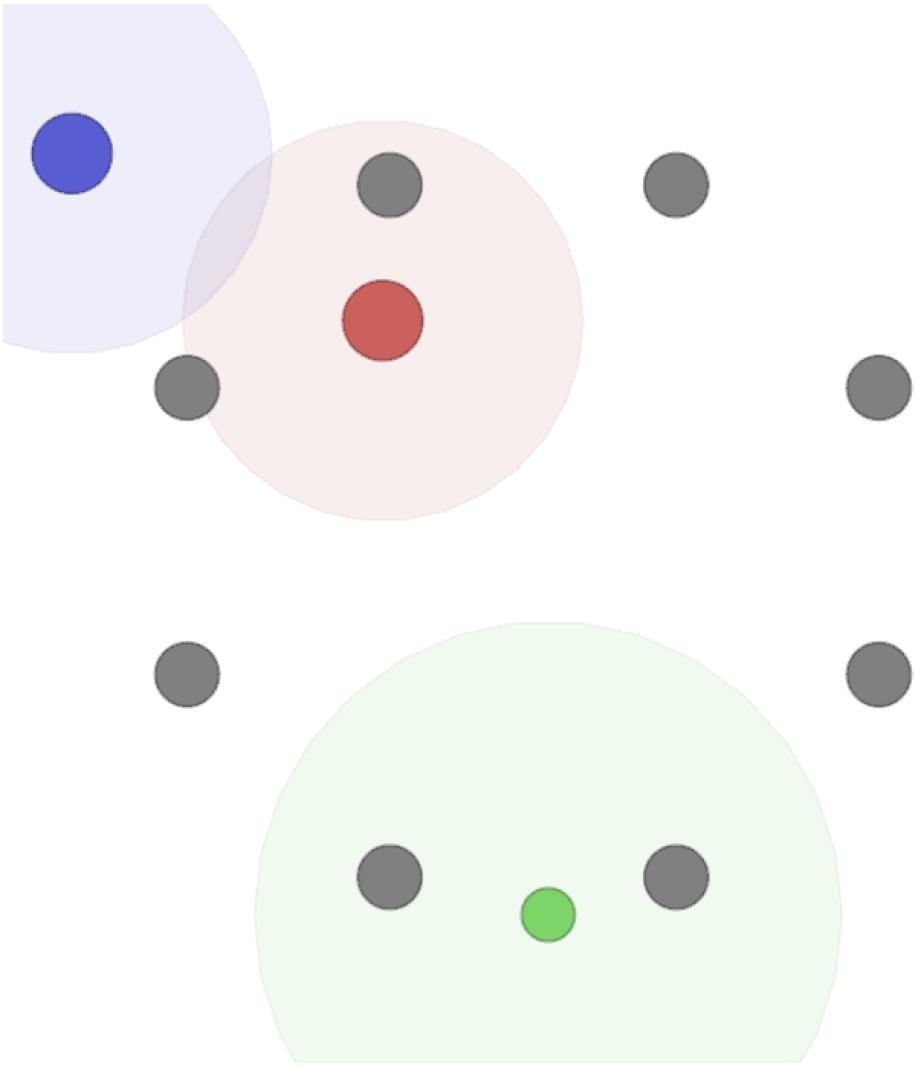

Figure 5: Visualization for Task 2.3

4 Task3: Reward shaping

In this section, I would like to explain how I set the reward shaping. As each actor can be learned by maximizing their own reward, reward shaping is very important task in reinforcement learning.

4.1 Task 3.1

I think there is only predefined reward for termination like when ranger catches poacher or poacher catches the animal. Therefore, I chose a method of giving penalties to spur each agent's actions if they do not in above two circumstances.

At this time, the method of granting penalties was also carried out with three main methods.

First, if neither Ranger nor Poacher were caught, it was a way to penalize everyone. This method seemed to be that the penalty was so overwhelmingly large so that learning was not going well.

The second method was using distance. The closer the distance between (ranger and power) or (UAC and power) becomes disadvantageous to the poacher, so a penalty was given to the poacher, and a reward was given to the ranger or UAC.

The third method came out to supplement the second method, and the second method found that the reward of the power continued to fall. Therefore, in order to solve this problem, reward shaping could be carried out by increasing the reward whenever the poacher collides with the landmark animal.

4.2 Task 3.2

I modified my code for reward shaping and upload them in Canvus.

4.3 Task 3.3

I train my implementation with reward-shaping for 300 episodes by using 3 different methods. I attach the figures like below.

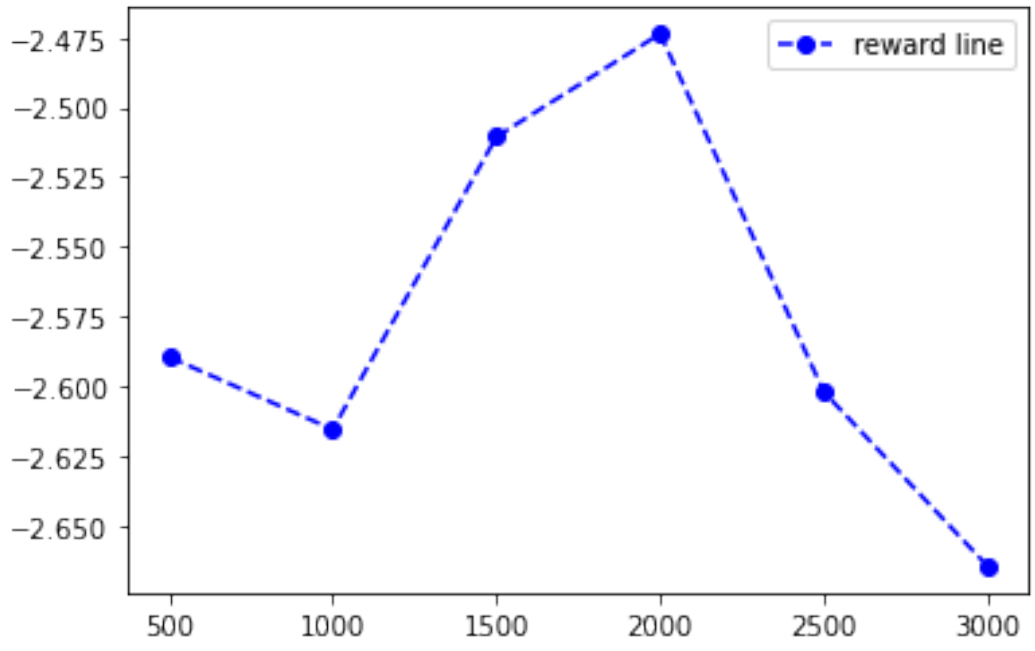

Figure 6: Reward shaping: penalty for Task 3.3

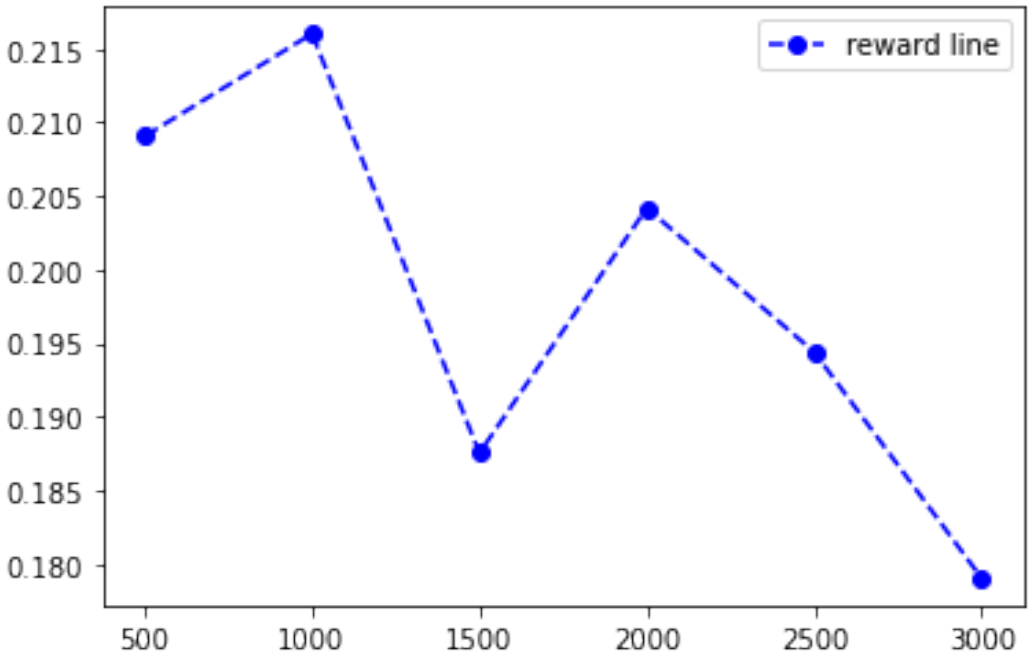

Figure 7: Reward shaping: distance for Task 3.3

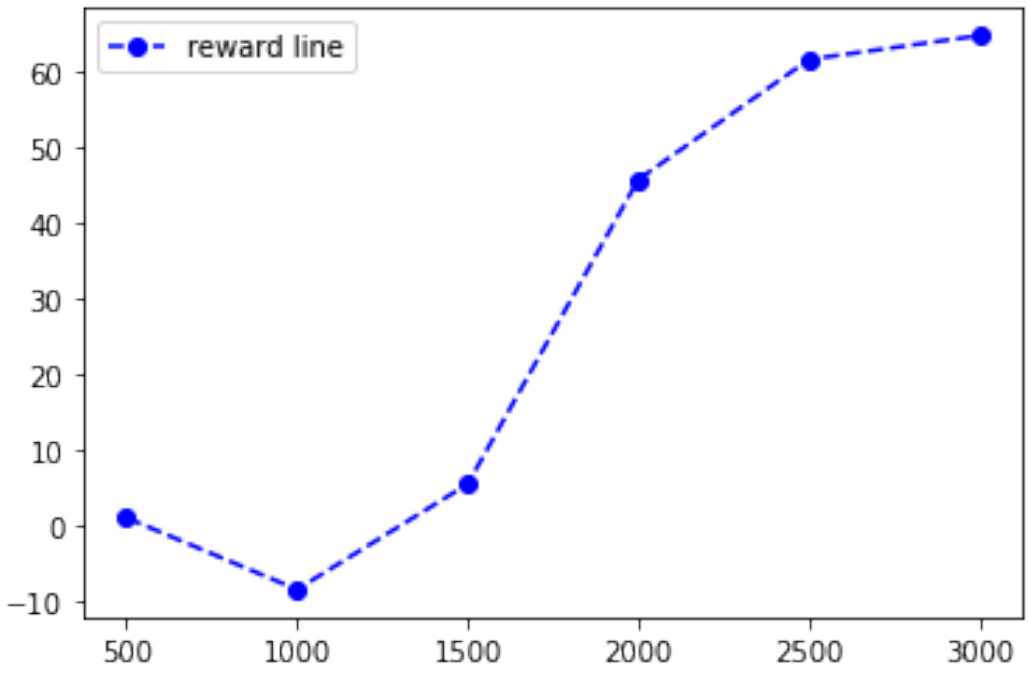

Figure 8: Visualization for Task 3.3

5 Task 3.4

There are question like "Does your reward shaping seem to help the agents learn better policies as compared to the case with no reward shaping? Why do you think this might be the case?".

I think that even if reward shaping is very import thing in RL methods, it needs to define very carefully. That's why I think my reward shaping is not better than before when seeing the gif file.

There are many case like when each agent do not take an action or it may be moving toward increasing the reward. It is very difficult to distinguish with them. I attach my gif file for reward shaping.

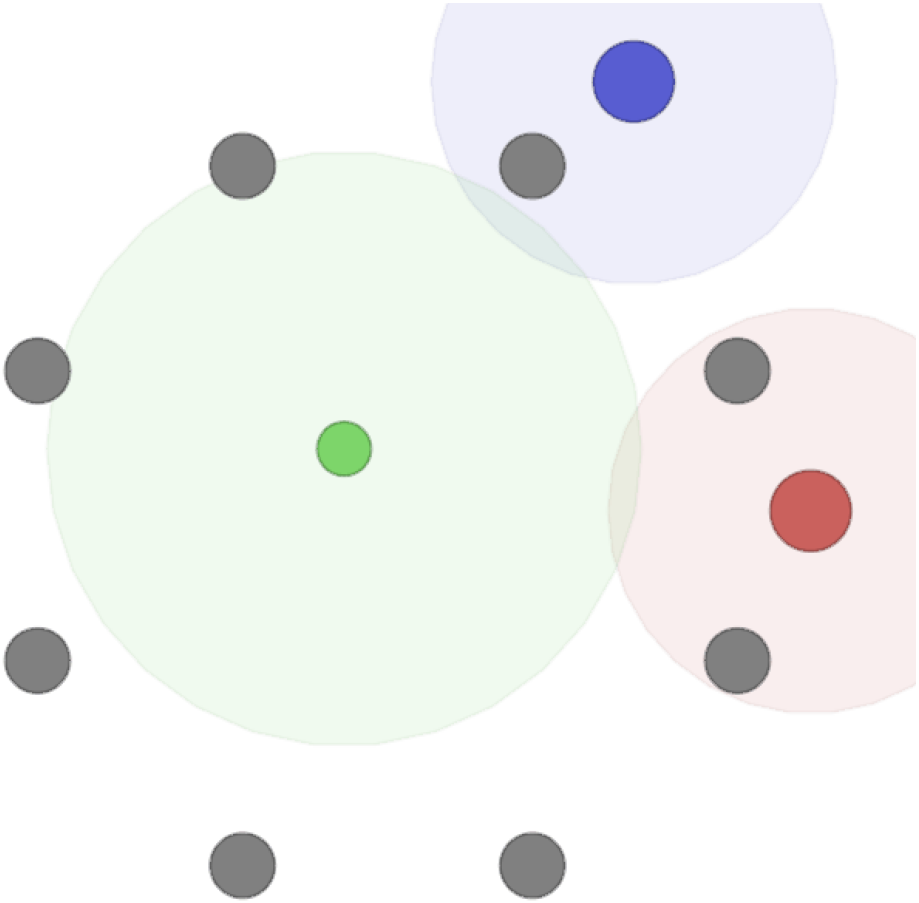

Figure 9: Reward shaping: distance for Task 3.4

6 Conclusion

In this homework, I can learn how to implement the code for MADDPG and how it is working. Plus, I can practice reward shaping, which is very important thing in RL. In the Future, I hope to apply reward shaping to cooperative system or centralized learning.