Reinforcement Learning on a Multiagent Coordination Game

Boura Tatiana
MTN2210

Sideras Andreas
MTN2214

Supervisor: G. Vouros

Problem Formulation

In this multiagent environment, exist seven agents that belong in two different categories, X and Y. Each category shows a preference to one the two available agent's actions. These agents are connected in a sparse manner via a connection graph and play a simple two-player coordination game, where if they both choose the same action, they get a payoff equal to 1. Otherwise, their payoff is equal to 0. The agents know neither the game they are playing, nor the available actions of the other agents. The purpose of this assignment it to implement a Q-learning, ϵ-greedy agent that learns how to play the game with the other agents he is connected to.

1 Implementation and results

The first multiagent environment we implemented, was the one seen in Figure 1

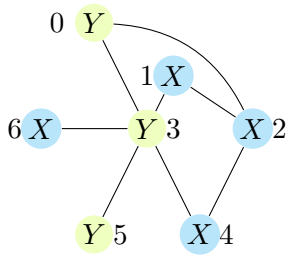
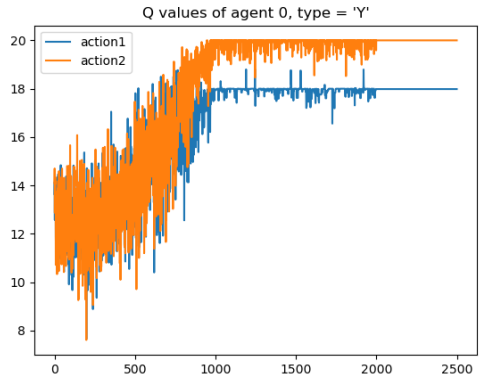
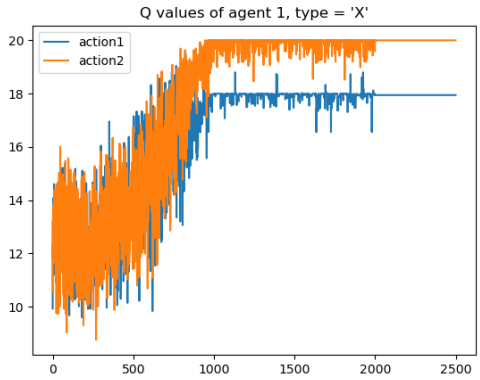


Figure 1: First multiagent environment

In this environment, since we are talking about a connected graph, we are expecting all of our agents to coordinate and play the same move. That is exactly what happened and all agents learned to cooperate. For example, in Figure are illustrated the Q values (the value of each move) in every episode for two different agents. See that, after at the exploration stage, i.e. until the 2000^{th} iteration, the value of each available action grows, but with oscillations. At the exploitation stage, each action's Q value converges to a value, making the move with the largest one the selected action.



(a) Q values of agent's 0 actions



(b) Q values of agent's 1 actions

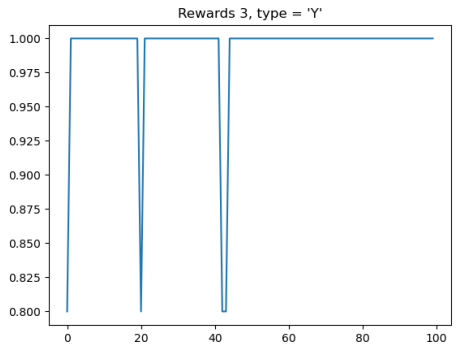
Figure 2: Q values of two agents' actions at every episode

Also, note that it is not random that the selected action was the second one. This is due to the fact that the agent 3, who is of type Y and is connected to five other agents, who he influences, prefers the second action.

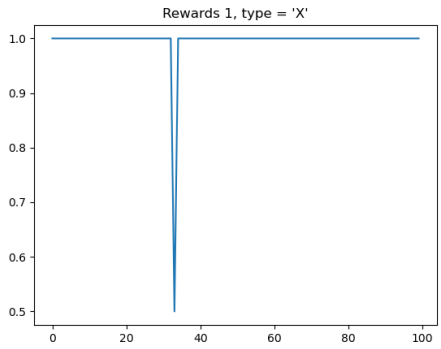
In Figure 3 we illustrate the average reward of two agent-types at each time step as computed with the following formula,

$$\overline{r}_i = \frac{reward_i}{\#neighbors_i},$$

where $reward_i$ is the reward the agent i receives at the current step given his selected action and $\#neighbors_i$ is the agents with whom agent i plays the game.



(a) Rewards of an agent Y at exploration



(b) Rewards of an agent X at exploration

Figure 3: Rewards of agents when in exploration

Note that we plotted the average reward at the 1000^{th} episode where the policy of each agent started to converge, thus we do not see many oscillations. If we were to plot the timesteps at the 500^{th} episode per se, then the average plot would look like something in Figure 4.

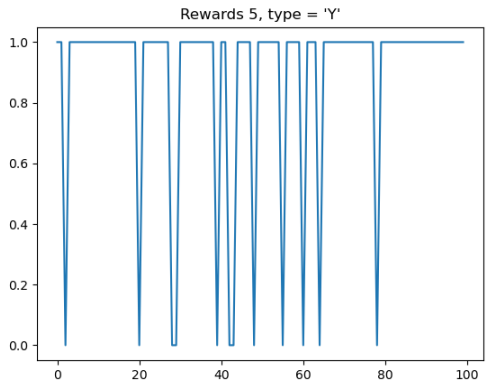
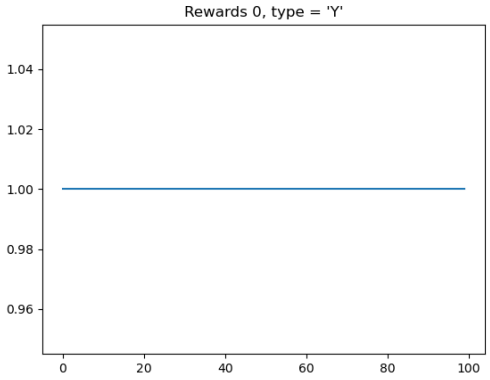
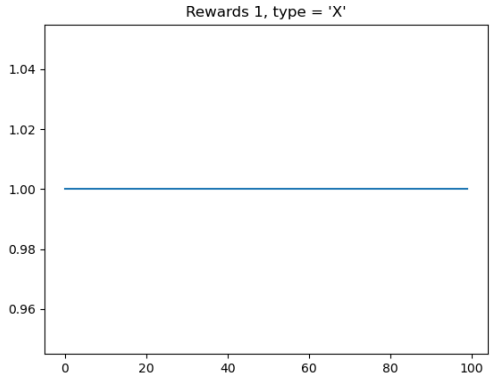


Figure 4: Rewards of agents when in early stages of exploration

Also, if we plot the average reward at the exploitation stage, where the policy has converged, the average reward equals to 1 at each time step, as the agents have learned to cooperate and get their maximum payoff.



(a) Rewards of an agent Y when $\epsilon = 0$



(b) Rewards of an agent X when $\epsilon = 0$

Figure 5: Rewards of agents when in exploitation

1.1 Other connection graphs

Now, let us consider the connection graph in Figure 6, that is not a connected graph. In this case we expect the agents 0,3,4,5,6 to coordinate with each other and the agents 1,2 to coordinate with each other as well. So, here the agents 1,2 learn to play the first action, which is expected as they both prefer it. The other players learn to play the second action, since it is the one preferred by the agent 3, who is connected to everyone.

For the connection graph seen in 7, one can notice that the only agent of type Y is the agent 5. This agent, has only one neighbor and thus influences the decisions of only one agent. So, in this graph all agents coordinate and play the action 1, that is preferred by most players and the agent 5 has to respect the decision of the other agents.

However, this is not the case for the connection graph in Figure 8. Here, as the only player Y is the only opponent of the other players, they all learn to play the action 1 (that he prefers) in order to coordinate with him.

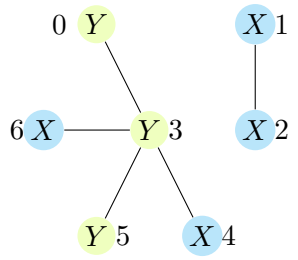


Figure 6: Second multiagent environment

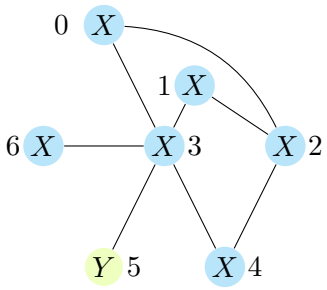


Figure 7: Third multiagent environment

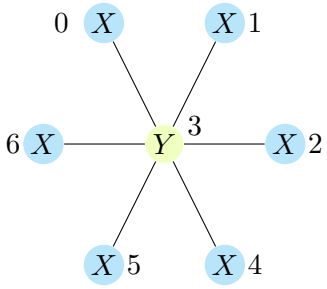


Figure 8: Fourth multiagent environment