Department of Electronic & Electrical Engineering
Faculty of Engineering Sciences

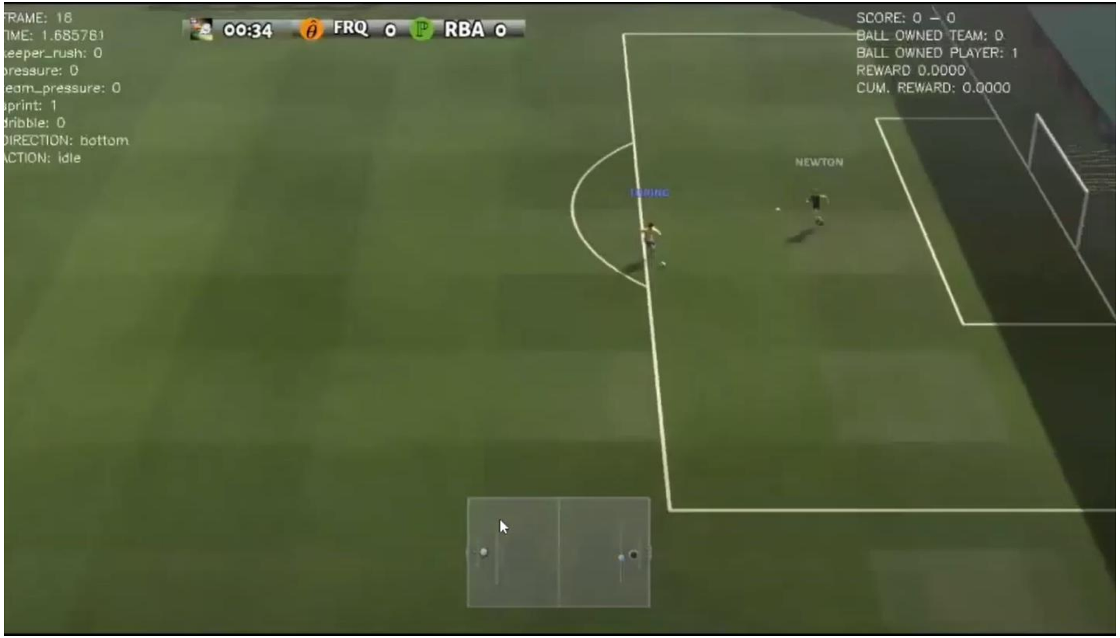
A Multiagent Cooperation Game Using Reinforcement Learning

Abdullah Alabdullatif

Supervisor: Prof. Wong

Introduction

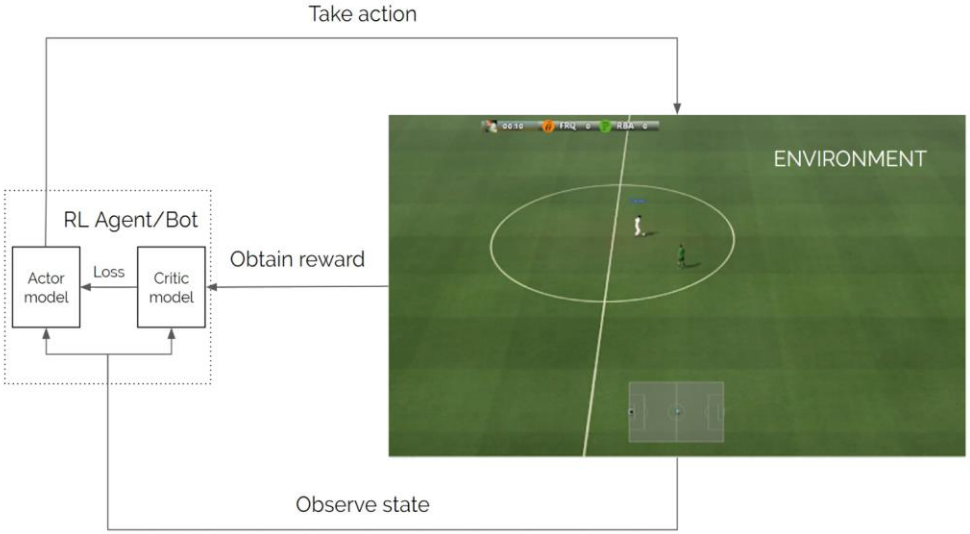
- The chosen game environment is Google Research Football, which was created for Reinforcement Learning research.
- The game contains different training scenarios, such as a full 11 vs 11 match, or even one attacker against a goalkeeper.
- Here is an example of a machine learning model that learnt to score in a creative way!



Reinforcement Learning Algorithm

- Reinforcement Learning is a branch of machine learning, where the agent observes the environment, takes actions, receives rewards, and learns through trial and error.

- The algorithm I used is the Actor-Critic, using a Proximal Policy Optimization (PPO) loss function.
- The actor model takes as input the state of the environment, and outputs an action for the player to take.
- The critic model gets the reward as input, and outputs a real number evaluating the previous action.



Experiments

- One of the main obstacles tackled by research is sparse rewards. In the game, a reward of +1 is given for scoring a goal. Another optional reward, called checkpoints, is added based on the distance to the goal.

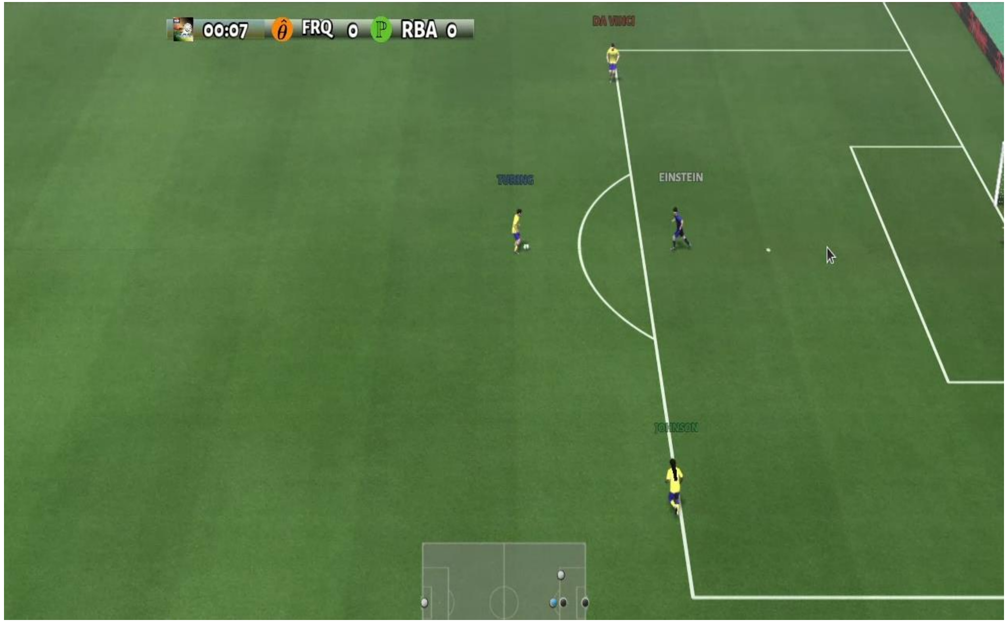
- Within the PPO algorithm, instead of evaluating an action based on the immediate reward obtained, it considers rewards obtained in the future. Actions taken closer to the reward are more significant.

- I trained models on an empty goal scenario with different reward setups (with and without checkpoints), and different algorithms.

- As for Multi-agent experiments, I trained a model to control three players against a defender and a goalkeeper.

Results

- The PPO algorithm performed better in the empty goal scenario as a result of considering future rewards. 1M steps of training was not enough to solve it if immediate rewards were considered.
- The PPO model with only scoring reward solved the empty goal scenario within 64k steps, as opposed to 1M steps in the research paper. As for the model with checkpoint rewards, it did not solve it within 1M steps.
- As for Multi-Agent training on three players, the model trained for 5M steps. They start taking random actions but then learn to pass the ball as shown in the video. They still do not score a goal.



Conclusion

- Considering future rewards to evaluate actions was an effective way to deal with sparse rewards.

- The Multi-agent model controlling three players was unsuccessful due to overfitting. The model exploits the checkpoint reward and does not try to explore other rewards.

- Similarly, in the empty goal scenario, the issue of overfitting caused the scenario to be solved within 64k steps, as the player only performs one action which is running straight into the goal, but not solved using the checkpoints reward.

Future work

- Short Term (Next two weeks):

1. Fix overfitting issue in my algorithm by setting a threshold maximum probability for actions to be chosen.
2. Continue training Multi-Agent model controlling 2 or 3 players and compare results to the research paper.

- Long Term:

1. Define custom reward functions such as passing and shooting on target.
2. Conduct Multi-agent experiments controlling more players such as a full team, which requires a lot of time and processing power.