Self Reflection

Student: Yu-Tse Ling - 1466145 – YUTSE

Part 1 Teamwork

The distribution of teamwork is equal as everyone is responsible for one agent with different techniques. What I contributed for the team is that I found resources of explaining the application of Monte Carlo Tree Search and Q-learning. The document and videos I collected for Monte Carlo Tree Search (MCTS) were straightforward and simple to understand, so we ended up choosing it as one of the techniques. We expressed our interests towards different techniques, so everyone was happy and willing to dedicate effort in it. My focus was on the MCTS agent, while the other 2 focus on heuristics and minimax individually. We set up deadlines every week to follow up the progress, while posting questions on WhatsApp group chat whenever difficulties appeared, so the problems were resolved quickly. In addition, I always added comments to the codes to make them understandable for my teammates in case they have thought of better add-ons to the codes for optimization.

Part 2 Technical Contribution

I was responsible for implementing the Monte Carlo Tree Search (MCTS) algorithm as one of our competing agents. It was not the best performing agent, but it got to win a maximum of 23 games, which is the second best agent in our team. It is an useful agent for running comparisons in experiments to help training and optimizing our best minimax agent to score higher points. Meanwhile, it is also a reference for our heuristics agent because the concept of sort_actions function can also be applied to the A* algorithm to aim for better performance.

Part 3 Written Contribution

The sections I wrote on wiki was (1) the problem analysis, (2) the part of the MCTS agent comparing to the heuristic agent in analysis, (3) running 10 games on the server and running games with the heuristic agent for experiments, and (4) the detailed description and explanation of MCTS agent implementation. We checked the content again together before finalizing the wiki to ensure that there were no mistakes and the written logic made sense.

Part 4 Learning

What I learned the most about teamwork is to communicate and provide suggestions proactively. By doing so, the discussions went efficiently without wasting time. Opening to opinions is also something I learned. Different people provide different insights towards the task, so I should not insist on my own ideas, but instead be openminded to comments, then try to organize the ideas to try on the existing agent. It might end up achieving higher scores.

Even after 12 weeks of learning AI planning, I still find it difficult and complicated to apply artificial intelligence to real-world tasks, but I am fascinated about how each algorithm works to reach the goal state and how powerful is can be. We have to design the agents with accurate and concise codings because if there are too many limitations, it impacts the effectiveness of the algorithm. In addition, defining the states, actions and rewards function correctly is crucial at the start as it helps debugging and the implementation won't be carried away through development.