Project proposition – multi-agent tank fight

Guy Carmeli 322988460

Nadav Raz 211873674

Sara Ifergan 322974320

1. what problem are we going to solve

The domain of our problem is a 2D tank game in which tanks are able to move forward, backward, rotate and shoot. While shelters are scattered in the map – so that a tank can avoid being hit.

The goal of our project is first to teach a single agent the mechanics of the game and be able to beat a human player – later after we trained the agent to know the mechanics of the game we want to teach the agents to work as a team. Making our second goal to find an influence model in which the agents work together.

In our domain definition the tanks know the entire map all the time (like humans while playing the game).

We assume that groups playing together will play better than single players/

2. how are we going to solve it

In order to solve our first task we will use 2 different AI approaches – which we have learned their basis in class.

The first approach is using DQRL – using a Q learning algorithm when the model is a neural network and not a Q table. We will give our agents meaningful features of the game based on the state – and that way we avoid searching for meaningful features based on the state image. We expect it to decrease training time.

The second approach we will try is using a genetic algorithm to train our agent – we will use a shallow neural network for our features as our mathematic model and use the genetic algorithm to update the weights.

After we will finish the first task we will have a project in hand as it meets all the requirements stated in the project instructions. Nevertheless we find this field of MARL very interesting and therefore we would like to expand our project to try to make a group of tanks fight together

In order to solve our second task, we will us RL. We will first teach them the mechanics of groups (not shoot your team members and aim at enemies). After we will have simple mechanics of groups we will try to explore different strategies – by sharing information about actions and rewarding agents who take other players actions in consideration and rewarding keeping the group safe.

In order to train our agent efficiently, we will build a heuristic AI agent which will compete our agent and help him to learn the basic characteristics of the game. Later, we will be able to train an agent against himself.

It's important to note that we are not entirely sure about the proposed solution for every task – especially the second one – we know that while researching and training we may change our proposed solution.

3. why we think it is the right way

For the first task we think that RL and genetic algorithm will be good ways for teaching our agents. We base our assumption on the fact that a deep Q network is a very strong model for reinforcement learning, and a genetic algorithm is a very versatile tool which can handle a variety of problems. We've checked online and found some people who managed to train an agent to play a game (not necessarily our game) using these 2 methods and therefore we believe we can do it as well.

For our second problem, we are far less sure that the proposed solution will work. Despite that, we did found a few articles claiming it is possible to find a suitable reward for teamwork social influence, and then to use deep Q learning in order to train the agents.

4. testing our results

Our first sanity check is to build a simple heuristic agent and check whether our AI can compete with it. We will also check it against ourselves. Another way of measuring the success of our agents is to look at the average score they achieve over a large variety of enemies and environments.

Checking the success of a group of tanks is similar. We can check it against single agents and compare the strategies.

<u>Demonstration</u>

A little glimpse for the mechanics of the game we wrote