CS5100 Project Proposal

Vineet Trivedi, Sharad Boni, Shail Shah

College of Computer and Information Sciences, Northeastern University

1. **Problem description**
2. *What problem are you solving?*

We aim to accomplish the following goal: Study the behavior of agents and determine the best agent in a stochastic, fully observable, dynamic, discrete, competitive, multi agent environment, when each agent is competing against the other agents.

We are going to build a multiplayer snake game, in which each snake’s aim is to eat food pallets without crashing into the grid boundary or a snake. A snake dies when his head crashes into a snake’s body, or one of the walls. When time runs out, the snake that is still alive and has the most points wins.

1. *Describe the problem formally from a computational perspective.*

Environment: N \* N grid, Agent, Food Pellets, Walls

Objective: Collect maximum food pellets without dying in a time frame.

Action: Agent can make 90 degree turns or continue moving in the same direction.

1. *What are the inputs and outputs (exactly)?*

Input: Actions as defined above.

Output: A new game state, determined by the action taken by each agent.

1. *Why is it interesting?*

We get to observe the behavior of 3 different algorithms in a competitive environment.

1. **Algorithms**
2. *What algorithms do you use?*

Q-Learning, Minimax with Alpha-Beta pruning, and A\*.

1. *Why are these algorithms appropriate?*

A\*: Searching algorithms that can search the game space to find the optimal path to the goal state.

Minimax with Alpha-Beta pruning: In a multi agent environment this is used to decide a move that maximizes the chances of winning.

Q-Learning: Reinforcement learning algorithm that will learn how to win the game.

1. *How are these algorithms typically used, and how are you using them?*

A\*: Used to go from a current state to the goal state using an optimal path when cost to move to the next state and heuristic to reach the goal state is provided. Example in robotics and graph search.

Minimax with Alpha-Beta pruning: Used to decide a move that will maximize the chances of winning in a multi agent game. Example, chess and Pacman.

Q Learning: Is a reinforcement learning algorithm used to update weights assigned to a list of features.

We are using the algorithms to search the game space to reach the goal state while competing against each other.

1. *Have other people use similar algorithms to solve your problem before?*

The algorithms have been used individually to win in this game format by other people. However, they have not been used to compete against each other in this game format.

1. **Results**
   1. *What results do you expect to show?*

We expect to show a comparison among the 3 algorithms and see when each algorithm works best by changing gird size, number of food pellets, time limit etc.

**References**

http://slither.io/  
https://becominghuman.ai/designing-ai-solving-snake-with-evolution-f3dd6a9da867  
https://towardsdatascience.com/train-ai-to-play-snake-in-your-browser-ca657097d707  
https://www.researchgate.net/publication/309390260\_Snake\_game\_AI\_Movement\_rating\_functions\_and\_evolutionary\_algorithm-based\_optimization