Student Name: Yunkai Wang
Student Number: 100968473
Project: Evolving neural networks using genetic algorithm to play the snake game

Description of the problem domain

Let the computer play a game is not a new technology, however, let the computer 'learns' to play a game is a relatively new topic and can have a lot of different usage.
For this project, I will train a neural network that plays the snake game using genetic algorithm, so that eventually the computer will become an 'expert' of the snake game. The snake game is a simple board game, the snake can only move in one of the four directions, so the neural network for the snake game will be relatively simple, the input neurons will take the current position of the snake, the position of the 'food' and the distances to its body or the wall as input, and return a direction using the output neuron. I will use genetic algorithm to train the neural network so that it will find some good parameters to use inside the neural network.

The motivation for the problem

The problem has appeared for a while, but this problem became popular since the Google's AlphaGo which beat the best several Go players in the world. This project is very interesting since the techniques can be applied to many applications with some modifications in the neural networks.
The general version of this problem, which is evolving a neural network using genetic algorithms has been studied and solved, the technique has been applied to a wild variety of applications like the heartbeat classification and game playing.

AI techniques to be used
Genetic algorithm:
Genetic algorithms are algorithms for optimization and learning based loosely on several features of biological evolution. There are five important components:
1. A way of encoding solutions to the problem on chromosomes
2. Evaluation function
3. Initial population of chromosomes
4. Operators that may be applied to parents when they reproduce to alter their genetic composition
5. Parameter settings for the algorithm

Neural network:
Neural networks are algorithms for optimization and learning based loosely on concepts inspired by research into the nature of the brain. They generally consist of:
1. A directed graph known as the network topology
2. A state variable associated with each node
3. A real-valued weight associated with each link
4. A real-valued bias associated with each node

5. A transfer function for each node which determines the state of a nodes as a function