IT3708 Project 3

Evolving Neural Networks for a Flatland Agent

Iver Jordal

# EA choices

## Parameters

|  |  |
| --- | --- |
| Adult selection method | Generational Mixing |
| Parent selection | Fitness proportionate |
| Adult pool size | 10 |
| Population size | 50 |
| Number of generations | 50 |
| Crossover rate | 0.5 |
| Mutation rate | 0.5 |
|  |  |

I used these parameters because they worked well on the previous project

TODO: Actually try different parameters and explain the process

## Fitness function

Each agent keeps track of how many items of food () and items of poison () it has consumed. The fitness of an agent (after its journey of timesteps) is computed this way:

In other words, there’s a high reward for eating food and a relatively small punishment for consuming poison. This allows the agent to explore more and potentially find and eat more food.

# ANN structure

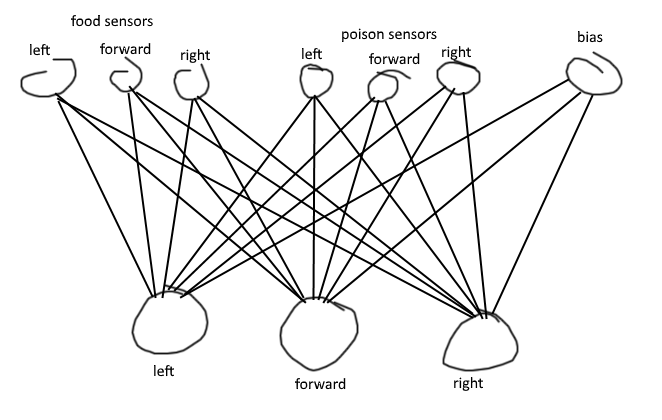


Figure 1: The structure of the artificial neural network

Because of the simple nature of the problem, I decided not to have any hidden nodes. However, I decided that I would need a bias node for those cases where no food nor poison is sensed. That allows the agent to have a default action for what it should do in those cases.

The activation function in the output layer is tanh and the threshold value for actually moving at all is 0. That is, if no output values are above 0, the agent won’t move.

The ANN is fully connected and the weights are in the range .

## The most successful agent

TODO: Show the neural network of the most successful agent and briefly explain its behavior.

## Process

TODO: Explain the process that I went through to find my design

TODO: Describe why my design should be able to solve the problem

# Performance of the EA

## Static run, single scenario each generation

TODO: plot static run and describe the behavior of the best evolved on this scenario

ASK: Should the board be the same in all generations or should it change from generation to generation?

TODO: test the best agent on a random new map