Project Proposal

Project Name: Neuroevolution with Multi-Agent Systems

Project Team Members: Jackson S, Jackson B, Kai S, Colton U, Adrian Q

Problem Statement:

Recent research on neuroevolution has shown that it is a “competitive alternative to learning by gradient descent in reinforcement learning tasks.” (Hamon 2023). This project will focus on continued research into applications of neuroevolution in reinforcement learning tasks, using simulation environments to evaluate multi-agent systems. We shall note the different behaviors between agents, such as “predator-prey” or “leader-follower” relationships.

Stakeholders:

Primary stakeholders: Prof. M. Ilhan Akbas
Secondary stakeholders: N/A

Proposed Solution:

A 2D simulation where collaboration or competitive behavior is required and each agent has a modular connection to a neural network brain. The neural network brain will be capable of Neuro-evolution. The UI will show stats about each agent, its brain, and the simulation itself. The user will be able to change certain parameters like activation function and simulation environment from the UI. The UI and AI backend will be modular and able to communicate in a clearly defined structure.

Things that are out of scope include the following:

1. 3D simulations

Notes:

- 2D environment (3D environment seems to be out of scope for this project and will not be considered)
- Group of agents can consist of “leader” and “followers”
- Group of agents will be placed in various environments; goal is to see how they adapt and work together from one situation to another
- Project set up to be modular; will be able to use different algorithms in different environments in order to see how each algorithm changes how groups of agents act
    - NEAT
    - HyperNEAT
    - ES-HyperNEAT
    - etc.

The following technologies shall be utilized for this project:

- Python
- Poetry (build system and package management)
- Rust…🦀
- Macroquad (Rust game library for rendering UI, used in GitHub reference)
- Cargo for rust build system and package management

Semester 1:

Semester 1 shall be spent primarily on creation and implementation of an interface for a two-dimensional (not just pixels; follows the look of the Rust project) simulation environment for the agents. This simulation environment shall be created using Pygame. A way to debug and/or change variables within the simulation environment shall be implemented. Likewise, an API shall be developed to facilitate the transfer of data between the AI model(s) and the developed simulation environment(s).

A very basic GUI shall be created for simulation testing purposes; this shall take shape in the form of a red* square actor in a white void, which shall be able to move in all cardinal directions.

A method of evaluating the fitness of agents shall be determined

Semester 2:

If anything is incomplete from Semester 1, tasks shall be rolled over. Nonetheless, Semester 2 shall be focused on the implementation of neuroevolution for the agents within the created simulation environment(s). The NEAT algorithm shall be implemented, with the ability to utilize / import(?) other algorithms later if desired.

Multiple simulations shall be able to run back-to-back; ideally, this would be done in parallel in order to facilitate the collection of data. The program shall be able to do so without having to be closed or reset.

Multiple simulation environments can be imported to be deployed into our application for testing agents.

Research more of how we should be re-using and re-training models after deployment into a simulation environment

Proposed Project Budget:

How much money is required to complete your project? Make a list of costs and what they are associated with, if applicable.

The project is not *currently* expected to require a monetary budget.

References:

Gautier, Hamon & Nisioti, Eleni & Moulin-Frier, Clément. (2023). Eco-evolutionary Dynamics of Non-episodic Neuroevolution in Large Multi-agent Environments. 10.48550/arXiv.2302.09334.

- GitHub - sparshg/asteroids-genetic: Interactive AI training simuation