***Project Proposal: Multi-Agents Simulation***

Hung Le, Elham Havvaei, Rouhollah Rahmatizadeh

***Description***: Implementation of real-time NEAT on a set of population. The agents wander within the environment, compete with each other for resources and try to survive as long as possible. Each agent has a limited amount of stamina which diminishes over time. The agent dies if stamina reaches zero. The agents can replenish their stamina by collecting food or receiving food from each other. The agent that gives food to other agents has a small chance to gain fitness. When certain conditions are met, agents with high fitness will be selected to breed for the next generation.

**NEAT** stands for NeuroEvolution of Augmenting Topologies. It is a method for evolving artificial neural networks with a genetic algorithm. NEAT implements the idea that it is most effective to start evolution with small, simple networks and allow them to become increasingly complex over generations. That way, just as organisms in nature increased in complexity since the first cell, so do neural networks in NEAT. This process of continual elaboration allows finding highly sophisticated and complex neural networks.

***Goal***: To observe a certain degree of altruism or cooperation among agents.

***Fitness function***: The fitness of each agent will be defined with respect to its lifespan and also the gained amount of food collected by the agent or granted to others.

***Domain***: Unity Game Engine ***Package***: Real-time NEAT C++

***Milestones***:

* Basic Domain setup (field, walls, food spots, spawn spot)
* Basic real-time NEAT setup (XOR test)
* Basic Communication setup (XOR test from Unity Engine)
* Advance Domain setup (agent inputs/outputs)
* Advance NEAT setup (generate population, fitness function)
* Advance Communication setup (connecting agents inputs/outputs with NEAT library)
* Evaluating, tuning and analyzing the behavior of simulated population
* Documenting the experiment and draw conclusion from analyzed data

***Tasks***:

* **Hung Le**: Setting up the domain:
  + Creating a play field using Unity game engine (in C#), which consists of:
    - A field, walls, food spots, spawn spot, agent (with inputs/outputs)
* **Elham Havvaei**: Setting up real-time NEAT:
  + Developing real-time NEAT to process inputs received from the game engine. For instance, computing the fitness of agents and obtaining the appropriate input for the evolving network
* **Rouhollah Rahmatizadeh**:
  + Setting up communication between Domain and real-time NEAT library
  + Reviewing the literature of enforcing altruistic behaviour by the agents
* **Shared tasks:**
  + Tuning parameters to obtain the desired result
  + Running the experiments, analyzing the collected data, and drawing conclusion
  + Writing report for the project

*Inputs*:

Each individual in the population will have:

* 7 rays casts out in form of a 120 degree cone
* Each ray cast would be wired to 5 input perceptron (3 for object types, 1 for distance, 1 for food level of agent in the same species if detected)
  + 000 means no object detected
  + 001 means a wall was detected
  + 010 means food was detected
  + 011 means an agent in the same species was detected
  + 100 means an agent in different species shelter was detected
* 1 perceptron for current food level
* 1 perceptron for amount of food give to other

*Output*:

* The output perceptrons will be wired with available actions
  + 4 perceptron for movement (forward, backward, turn left, turn right)
  + 1 for give away food
  + 1 for picking up food