## *Training Genetic AI Model on Grammatical Evolution for Creating OpenTTD AI Scripts*

#### *Ravin Aziz*

#### *2689672*

## Proposal

### Motivation

*Games can be thought of as forms of simulations that aim to mimic aspects of reality and real-life, one such game is OpenTTD. OpenTTD is a game with the objective of developing a robust logistics and transportation company that can outperform competing companies. The implementation of OpenTTD as a reflection of real-world scenarios, primarily as a medium to assess the feasibility and efficiency of establishing a public transportation network, offer us the opportunity to train an AI model in an inexpensive but highly scalable environment.*

### Aims

*This project aims to train a Genetic AI model on grammatical evolution to create different in-game AI scripts. The scripts are then tested on multiple different parallel seeded maps for a certain period of in-game time to evaluate the overall performance in different situations. The metrics measured for performance of the script will be based on the information provided by a save-game parser to which subsequently utility functions will be made to account for different priorities and considerations. To which the results will then be synthesized and visualized on graphs.*

## Progress

* Wrote an AI script focusing on creation of public transportation (bus) networks in, Squirrel, derived from an existing and popular AI script, SimpleAI
* Integrated testing and evaluating the performance of the AI scripts using the OpenTTDLab library and its experiment function.
* Determined the 3 utility functions to evaluate performance (with potential for more): profit, cost, and waiting time
* Parameterized the AI script source code and determined key variables
* Defined the value constraints of the parameter values
* Integrated OpenTTDLab with the DEAP library for GA AI training.
* Used the defined utility functions to train the GA AI model for parameter tuning on the custom game script, each utility function running on roughly ~20 generations

## Problems and risks

### Problems

* Running the experiment function from OpenTTDLab requires constant internet connection because it uses the socket function in python
* Implementation of the main while loop in the AI script is not robust enough and can influence the final results
* Highly dependent on the OpenTTDLab save-game file parser, lack of customizability of metrics.

### Risks

* Potentially difficult to determine the encoding from AI results to the grammar. Mitigation: will do further background research and testing on existing methodology and approaches
* Training the model might take a long time due to the constraint of the parameter values or the constraints might be incorrect in the first place. Mitigation: will have to experiment with the game more to establish broad but reasonable search space

## Plan

**November - December**

- Finish everything to do with the data needed in the jupyter notebook

- Get access to the servers before the holiday and run the test for bigger population and higher generations

- Start reviewing grammar definitions

- Review source code for the last times (making adjustments as necessary)

- Start drafting the report

**January**

- Start drafting the grammar definition

- Continue working on the report

- Finish the grammar definitions (Hopefully before the end of the month)

**February**

- Finish the grammar definitions (If not yet done)

- Run and train the model on the servers

- Process, collate, and organize all the data from the servers

- Continue working on the report

**March**

- Continue working on the report and finish up

- Send draft preferably 2-3 weeks for review to supervisor before deadline