Project Outline – Evolving Creatures (Genetic Car Simulator)

By Jonathan Brooks (job64@aber.ac.uk)

03/02/2019

V0.3

Description

The project is to create an evolutionary algorithm for a pre-existing JavaScript framework. The pre-existing framework is a 2D car racing simulator [1] where cars race over several instances. Each instance Is a new population generated via an Evolutionary algorithm using the previous instance population.

The project requires the analyse of the code for this Genetic Car simulator to understand where I will be adding my own additions to the code and understanding the data/car format which my Evolutionary algorithm will be manipulating to create new Car populations to run the simulation.

An Evolutionary algorithm in general optimises a pre-existing solution to work better against a given criteria such time complexity. An Evolutionary algorithm has a couple of main components such as Selection where a Parent/Object Is chosen based on some criteria, Mutation where a random change happens to a child object's genetic code[3].

Several iterations of my own Evolutionary Algorithm will be created and improved over the course of the project, which will be put into the existing framework and tested. A classifier i.e. Clustering will be introduced as an extension of the selection process where the criteria by which a parent is chosen changes as the Algorithm runs.

I will be conducting an Empirical analysis over the different iterations of the algorithms comparing them with a baseline Evolutionary algorithm which is commonly used. This will be used in the testing process to help make rationalised changes to the Evolutionary algorithm.

The agile methodology that will be used in the project is Scrum which will provide a good Overview and timeline of the project while allowing for the flexibility to make incremental non-planned changes.

Proposed tasks

(07/02/19 – 21/02/19) – The framework is written in JavaScript which I have little experience in, so I will be learning more about the language through the course of the project.

(07/02/19 – 21/02/19) – For the start of the project I will be going through the framework trying to find which parts of the framework are the genetic algorithm code and what parts of the framework interact with the Algorithm. This will allow me to know where I should be replacing and connecting code.

(14/02/19 – 04/04/19) – Testing and Design documentation, the design documentation will consist of what within the framework was changed and how the overall algorithm works and interacts with the framework. The testing will have the unit testing.

(14/02/19 – 25/04/19) – Project Final Report

(14/02/19 – 21/02/19) – Introducing a simple stripped-down evolutionary algorithm replacing the pre-existing algorithm. This will test the interaction points between the framework and the new Evolutionary algorithm, such as returning new population to the framework.

(21/02/19 – 04/04/19) – Additions to the Evolutionary algorithm will be included such as Selection, Crossover and Mutation. These parts would be individually tested and tested together analysing the performance of the Cars, while making incremental changes to the algorithm to improve performance. This will be an ongoing process through the project.

(21/02/19 – 07/03/19) – A second Evolutionary algorithm alongside my own implementation which will be used as a comparison point.

(07/03/19 – 04/04/19) – The GUI/html side of the framework will be changed to include statistics from the evolutionary algorithms which can be used for comparison. This will be expanded on so that the user can view results over several instances. The framework would need further exploring to implement the changes to the GUI.

(07/03/19 – 04/04/19) – Researching Classifiers and Introducing a classifier in the selection process of choosing the parents, and possibly introduced in the crossover and mutation process.

Project Deliverables

- The framework should successfully use an Evolutionary algorithm in combination with a classifier to create new populations of cars for every instance of the simulation.
- The implemented evolutionary algorithm should also optimise the cars performance over every instance run, getting better the more instances that are run.
- The GUI would show statistical results for all run instances.
- The user would be able to compare the statistics of different instances past and present.
- The statistics would show the results from both Evolutionary algorithms, my own implementation and one for comparison.
- Testing documents would consist of unit tests to test the functionality.
- Design documents would give an overview of the algorithm and how it works.

Bibliography

[1] Rafael Matsunaga, "HTML5_Genetic_Cars", 2013 [Online], Available: https://github.com/red42/HTML5_Genetic_Cars

- Framework that is being used as the bases of the project.

[2] Khalid Jebari & Mohammed Madiafi, "Selection Methods for Genetic Algorithms", December 2013 [Online], https://www.researchgate.net/publication/259461147_Selection_Methods_for_Genetic_Algorithms

[3] "Genetic Algorithms Tutorial", [Online], https://www.tutorialspoint.com/genetic_algorithms/index.htm

[4] "The Modern Javascript Tutorial", [Online], https://javascript.info/

Change Log

03/02/2019	Job64	Created Document, Added Description, Added Proposed Tasks and Project Deliverables.	V0.1
03/02/2019	Job64	Added Cover Page and Change Log.	V0.2
07/02/2019	Job64	Added Bibliography and References, Changed Description.	V0.3