Cellular Automata in the A* Plane

Matt Trefilek
Advisor: John Sarkela

Project Proposal

The goal of this project will be to create a Model-View-Controller application of simple automatons navigating a hexagonal plane. The hexagons will be a data structure containing information about their locations and the addresses of the sides. The two addressing methods allow for paths to be represented both by the hexagons the automaton travelled through and an abstract shape created by the edges traversed. The hexagons will then be joined in a larger two-dimensional plane which will be the space that the simulations are run in.

I will also create automatons that operate within this space. The space will have a number of attractors and repellers. The automatons will be structured similar to Braitenberg vehicles, containing sensors and actuators. As the vehicles move through the space, they will exhibit nonlinear behavior based on the rules governing their sensors interaction with the attractors and repellers.

The program itself will have an A* space with a number of defined entities. A viewer built in Java Swing will be used to track the paths of the automata as the simulation progresses. The viewer will also offer tools for editing and running the simulation. The viewer will be an observer to the underlying data model and control layers. The application will be used to run the simulations and analyze the paths that the automata take around the different plane configurations. Tools will be available for the analysis of the paths chosen and the interactions between the automatons and the attractors and repellers.

Technology

The Data Model and the Controller portion of the project will be written in Java. I have chosen Java for several reasons. It is an Object Oriented Language, it has many frameworks that fit my needs for graphics and visualization, and I want to gain a deeper knowledge of project management and unit testing with Java.

The View portion of the project will be created in Java Swing for the 2D components, implementing the Observer Pattern between the Model/Controller and the View. The 3D portion will either be written in Java using OpenGL, Unity using Javascript, or Blender using Python.

On the project Management side, I will be using git for version control and learning JUnit4 for the unit testing. I will also be adhering to the Object Oriented Programming Principles.

Grading Rubric

Rubric:

30pts: Research
- 5pts: Advanced Knowledge of Java Swing
- 5pts: Understanding of the A* Plane and Pathing Methods
- 10pts: Research into Cellular Automata and Braitenberg Vehicles
- 10pts: Research into Nonlinear Dynamics and Attractor Processes

10pts: Develop an Data Storage and Addressing system
- 7pts: Data Model Implementation
- 3pts: Abstraction of Paths from Data Model

25pts: Pathfinding Automata
- 15pts: Pathfinding Utilizing Attractors and Repellers
- 5pts: Recovery of Individual Tracking Information
- 5pts: Recording Attractor and Repeller Basins

45pts: 2D Visualization Engine
- 10pts: MVC Design - Using Data Model and Pathfinding Controllers from Above
- 5pts: Swing GUI with Simulation Controls
- 15pts: A* Canvas
  - 5pts: Representation of Hex Grid
  - 10pts: Automata Tracking and Displaying Paths
- 5pts: Implementation of Observer Pattern
- 10pts: Results View: Map of Attractor and Repeller Basins

10pts: Analyzing and Displaying Examples of Different Behaviors that Emerge

10pts: Project Management
- 2pts: Git Version Control
- 2pts: Object Oriented Project Management
- 6pts: Unit Tests with JUnit4

Total: 130

Scale:

A: 110-130
B: 90-110
C: 70-90
D: 50-70

Conclusion

This project is an exciting opportunity for me to combine my math and computer science education. I am excited to learn more about a number of different frameworks and concepts while also increasing my knowledge of industry tools.