Alex Johnson
CS4300 - Graphics Project
Progress Report

Goal:

To develop an experimental visualization of swarm behaviour. This passive visualization will provide an visually engaging environment to demonstrate swarm (flocking) behaviour. This project was inspired by Craig Reynold's work in creating algorithms to emulate the navigational behaviour of creatures such as birds and fish. I find the concept of organic emulation such as swarm behaviour to be fascinating. While movement appears random, every action is determined by the surroundings. The following quote is from Craig Reynold's online writings on boids.

> *"A significant property of life-like behavior is unpredictability over moderate time scales...It would be all but impossible to predict which direction they will be moving (say) five minutes later…. This property is unique to complex systems and contrasts with both chaotic behavior… and ordered … behavior. This fits with Langton's 1990 observation that life-like phenomena exist poised at the edge of chaos." (source)*

I plan to expose the underlying behaviour, and allow parameters manipulations for users interested in the algorithm. However, for most users, I hope that this piece could stand alone as a pleasing graphical visualization.

Themes:

Games as Art, 3D Visualization, Visual Analytics

Accomplished:

Progress can been seen in the github repository:
alexjohnson505/swarm-navigation

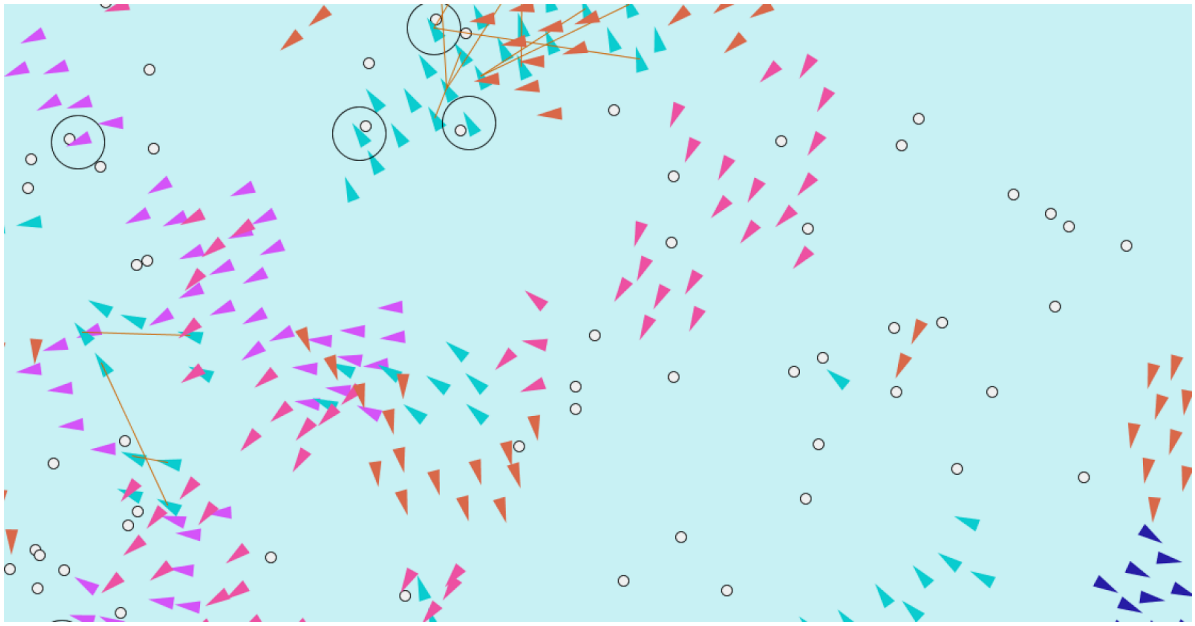
I've uploaded a development version of the project to my website:
http://alexjohnson.io/swarm-navigation/

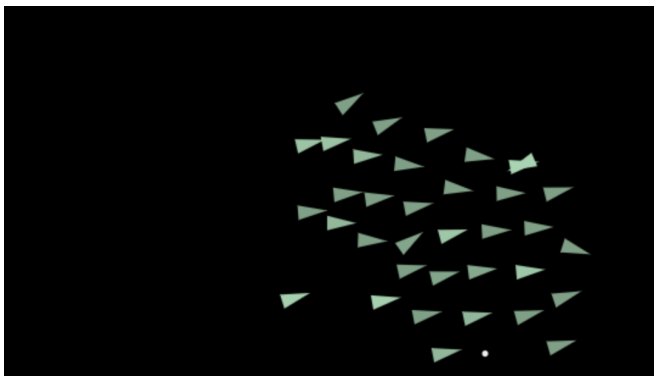
Accomplished features include:

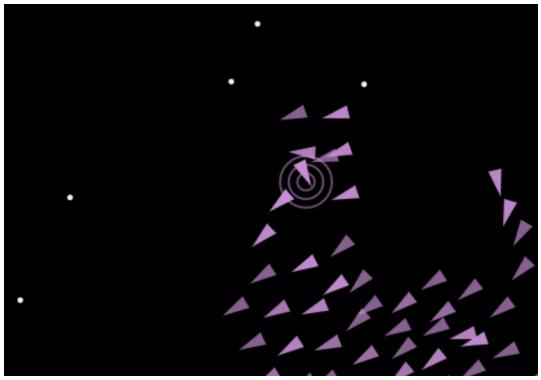
- Rendering a live canvas utilizing Processing
- Implemented Craig Reynold's Swarm algorithms. De-contructed, and refactored the Javascript from Daniel Shiffman's book Nature of Code.
- Converted Processing project into Processing.js webpage.
- Added support for multiple swarms.
- Added support for "survival of the fittest" - an experimental mode where swarms compete against each other and starvation for limited re-spawning resources.

Screenshots:

The following screenshots show different stages of experimentation from the development process:







To-do List:

An up to date list of tasks can be found in this project's github repository. Key features include:

User Interface:

- Provide User Interface controls for allowing the user to initialize and configure the environment. Users should be able to tweak the initial setup, and personalize the starting variables to experiment with the algorithm.

Refactoring:

- Improve data structure for efficiency. The current boids algorithm compares the proximity of every member of the swarm. Since members are only influenced by their immediate neighbors, we can organize the data structure by proximity, and this drastically reduce the amount of computation at higher population levels.

Sandbox Mode

- Create a "Sandbox Mode". Part of this project revolves around the exploration of flocking algorithms. I'd like to create a mode under which users could visualize the variables of flocking (proximity, direction), and adjust the global parameters for a swarm. These parameters include settings such as cohesion, swarm size, and turn radius.

Engagement Mechanisms:

- "Survival of the Fittest". As a visually engaging example of swarm mechanics, I'd like to expand on a multi-swarm competitive mode. This mode could demonstrate the efficiency of different swarm configurations. These mechanisms could introduce aspects such as swarm competition, resource competition, and the possibility of predators.

"Stretch" Goals

- 3D Rendering. I'd like to expand Craig Reynold's original algorithm, and support a 3rd (vertical) dimension of travel.
- Migrate to native Javascript. The current implementation us using an interpreted processing.js file type (.pde). I'd like to migrate the project into native Javascript.
- Alternate Flock AIs. While Craig Reynold's original algorithm provides a great base of example, there are a few other swarm algorithms that I would like to compare and visualize.