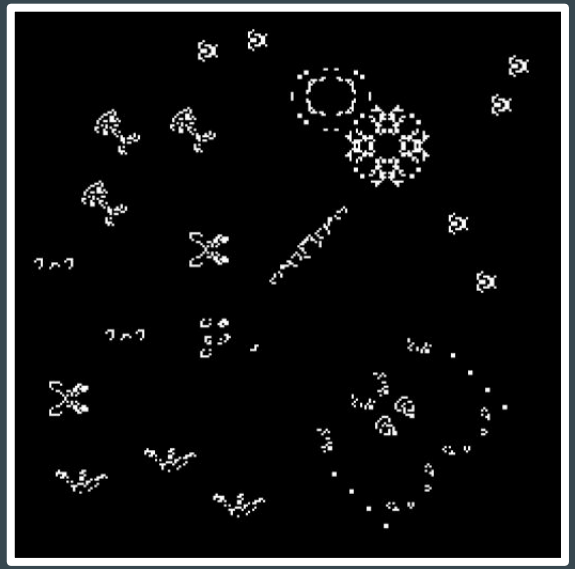
Simulating Learning in Generated Environments

Austin Barner, Vihan Garg, Zakk Loveall

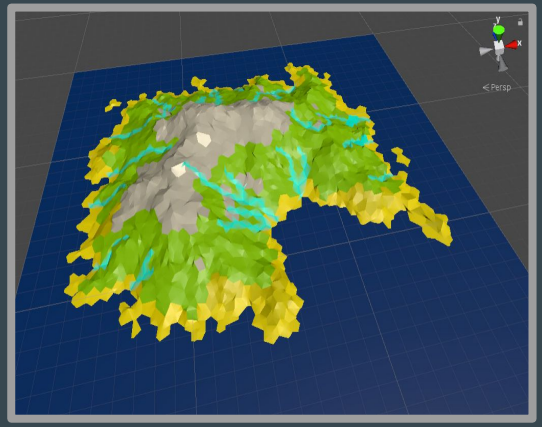
Background

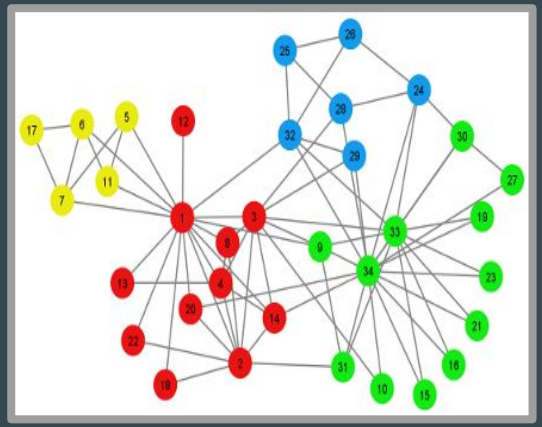
- Wanted to simulate creatures in environments
  - Similar to Conway's Game of Life
- Needed to break our problem into pieces
- Realized one goal was creature behavior
  - Movement algorithms
- Another goal was world generation
  - Creatures need environments to traverse
  - Learning to traverse unique environments



Implementation

- Decided on Unity to do our project
  - Provides UI
  - A lot of resources
- Split our project into two parts
  - World generation
    - Mimic an actual environment
  - Meaningful movement behavior
    - Mimic creature learning





World Generation

- Utilize Open Simplex Noise
  - Derived from Perlin Noise
- These algorithms allow for ‘natural’ looking terrain generation
  - N-dimensional
  - Pseudo-random
  - Seed-based
- These algorithms are commonly used in video games
  - Minecraft

Example perlin noise vs random noise map:

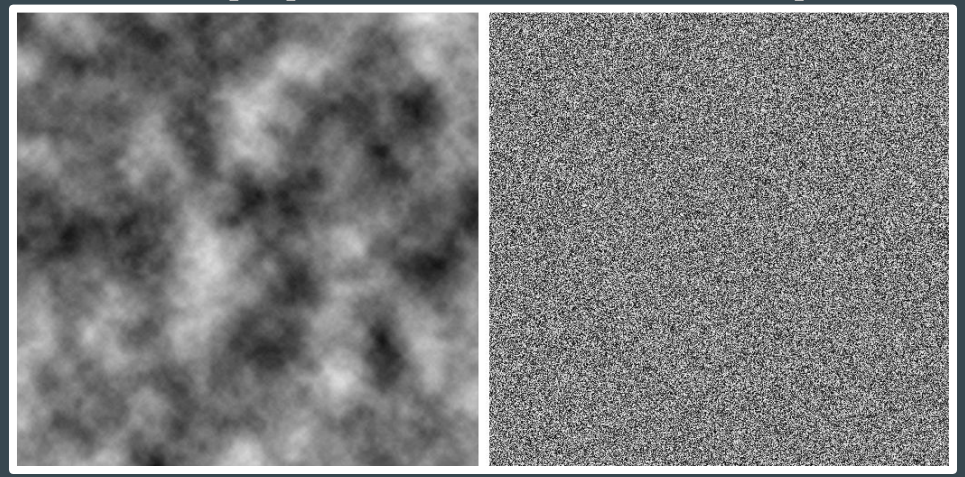
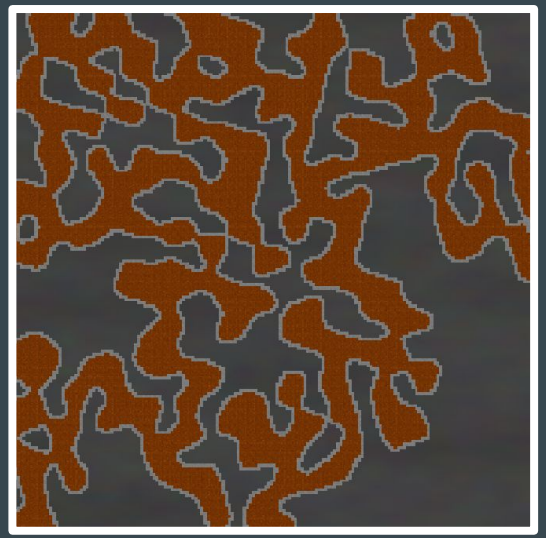
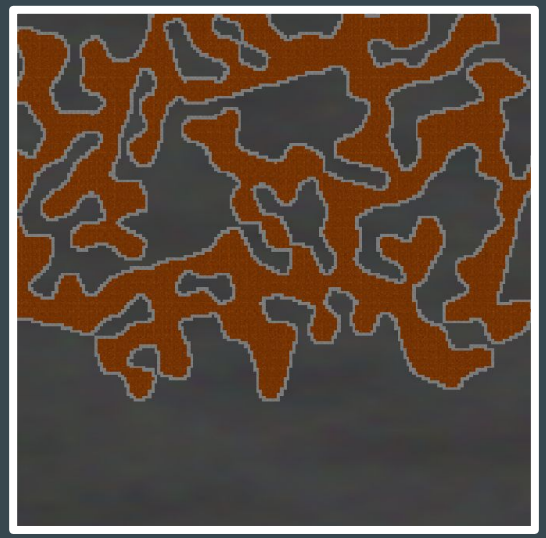


Image source: https://brandenstrochinsky.blogspot.com/2015/05/random-map-generator.html

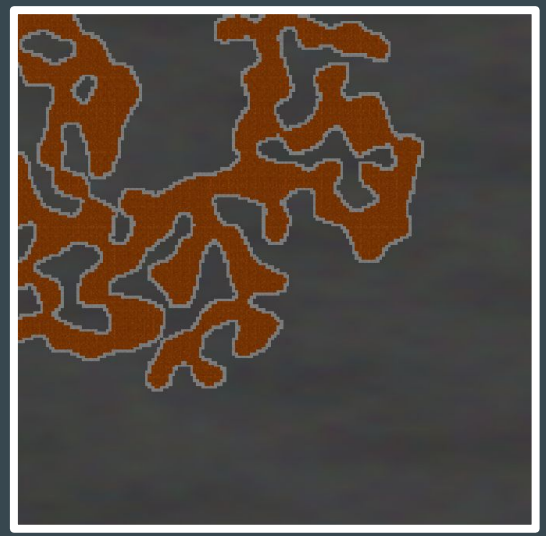
World Look and Feel



Seed: 1805235358



Seed: 317055230

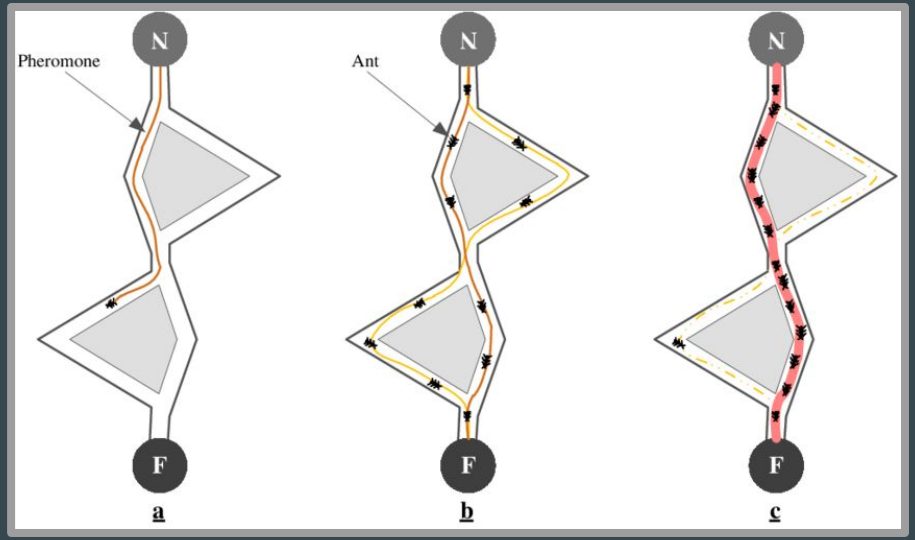


Seed: 564933020

Open Simplex Noise Algorithm: https://gist.github.com/digitalshadow/134a3a02b67cecd72181

Movement Algorithm

- Goal was to simulate learning
- Ant Colony Optimization
  - Simulate learning behavior
  - Good for finding paths through graphs
    - Our terrain was a glorified graph
  - Give the ants food source to work towards

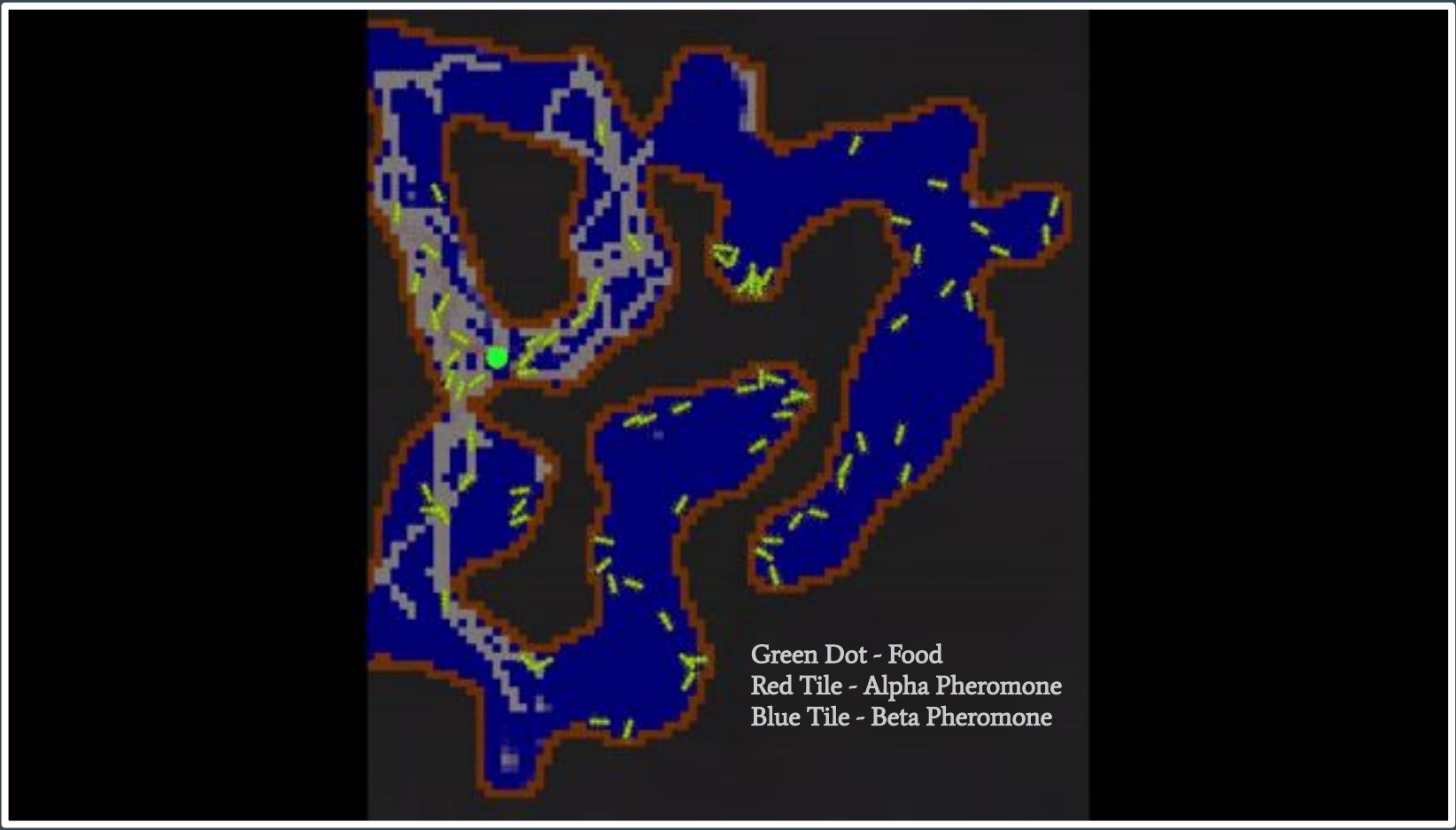


Steps for Implementation

- Drop Pheromones
  - Pheromone strength associated with tile
- Dual pheromone system
  - Avoid clustering
  - Encourage exploration
- Following pheromone trails
  - Ants will follow pheromone trails
  - Alpha evaporate over time
- Find Food
  - Use alpha pheromone trail for food location
  - Use beta pheromone trail to determine density

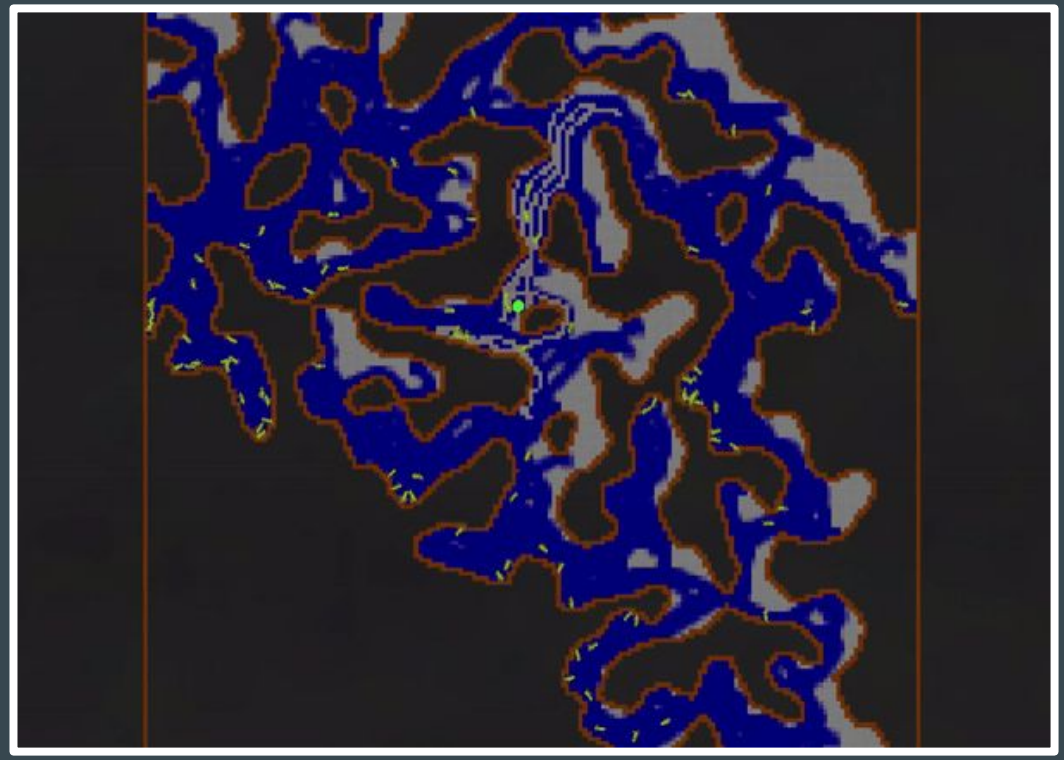
Image was cropped

Ant Movement Demonstration



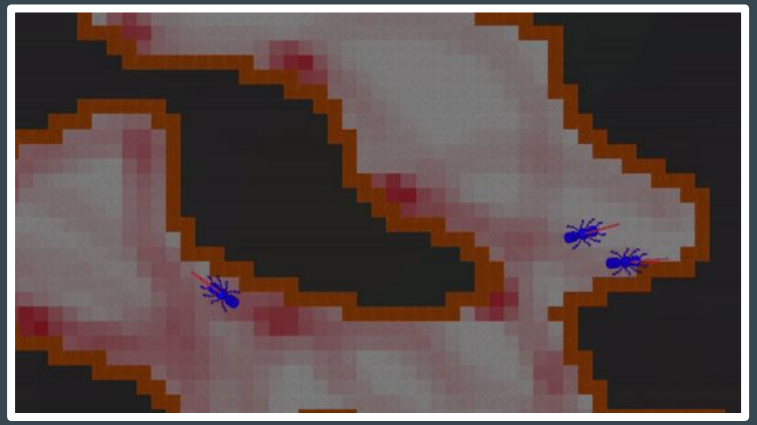
Challenges

- Team Challenges
  - Defining scale/scope
  - Finding best solution to problem
  - Agreeing on ideas
- Search Algorithms
  - A* search algorithm
  - Neural network based learning
  - Genetic learning
- Many issues with these
  - Too simple
  - Too complex
  - Not applicable



Future Work

- Learning-based simulations
- Potential games to be made
  - Tower defense games
  - A life simulator game
- More complex behavior
  - Fighting or foraging
- More complex creature structure
  - Evolves based on weak points
- More complex environments
  - Add water or trees
- Completely fix clustering problem
  - Improve beta pheromone handling



Conclusions

- Learned a lot
  - Compromise
  - Trial and error
- We can mimic evolution-based learning
  - Ants were able to follow pheromone trails to food
- Our generated environment did work
  - Environment simulated what we wanted it to

Sources:

- http://www.ludowaltman.nl/slm/
- https://gamedev.stackexchange.com/questions/173190/how-to-make-procedurally-generated-terrain-look-realistic
- https://www.geeksforgeeks.org/search-algorithms-in-ai/
- https://en.wikipedia.org/wiki/Ant_colony_optimization_algorithms
- https://experiments.withgoogle.com/conway-game-of-life

Questions?