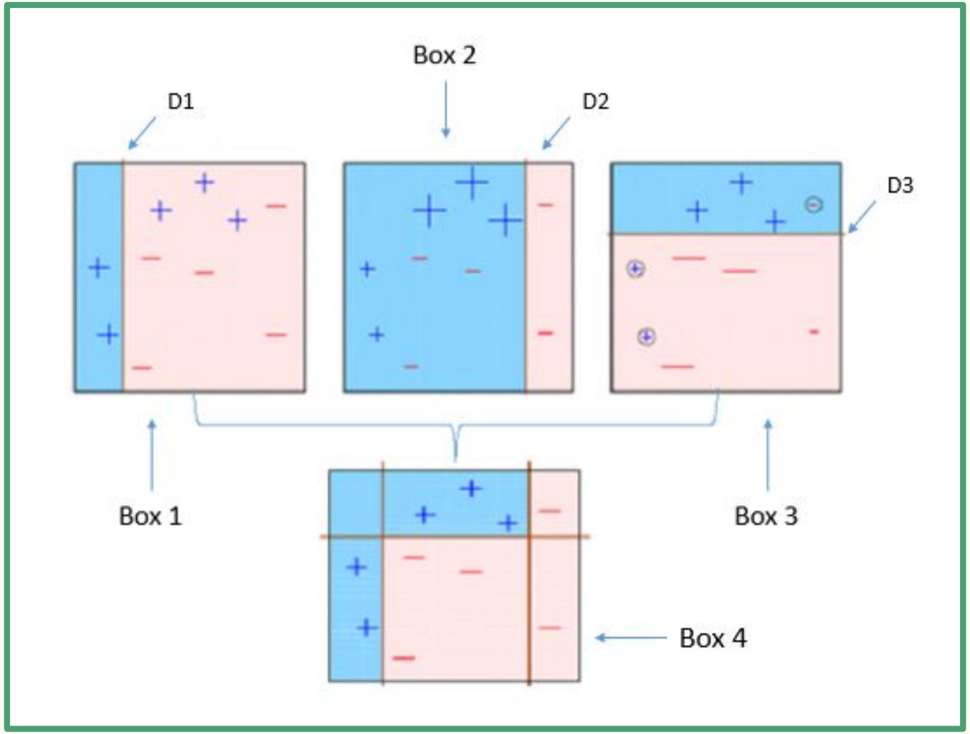
Conveying Directional Information Through Regional Division in Swarms

Chris Cedron
Clayton Dembski
Floris van Rossum

Introduction

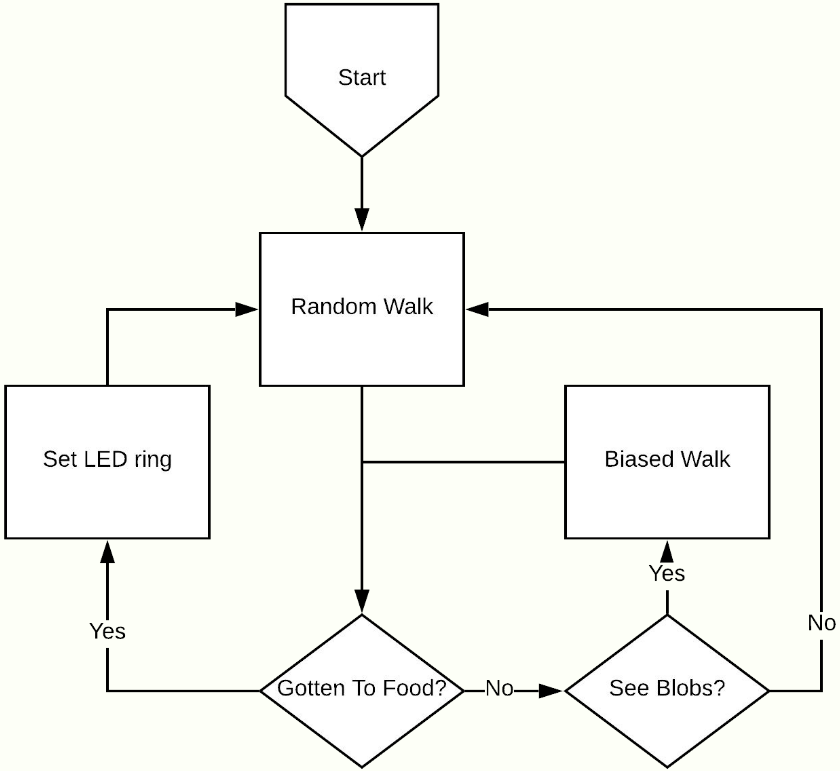
Adaptive boosting

- Take multiple weak classifiers and combine them into a strong classifier
- Each classifier divides space into two regions
- Regions are aggregated to create robust divisors



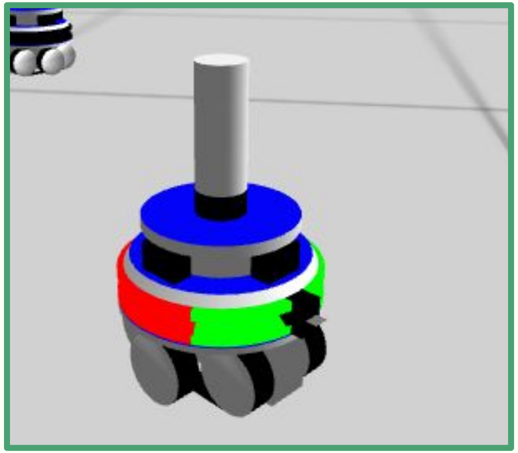
Premise

- Robots only communicate with the LED ring and camera
- Robots try to converge at a specified point somewhere in the world
- Each robot that knows divides the world into two regions about the point of interest



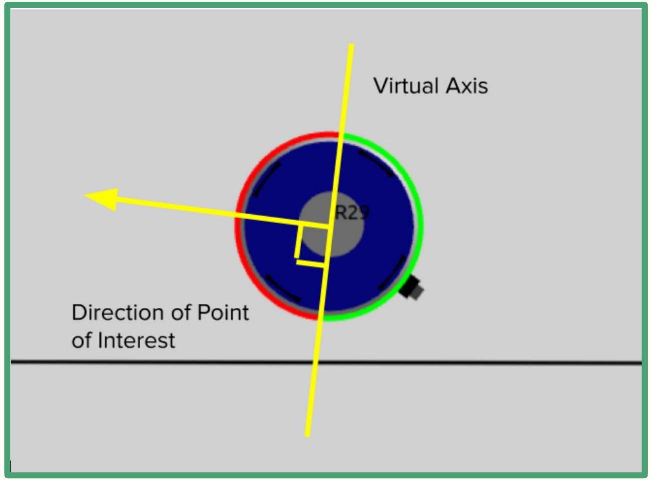
Methods and Theory

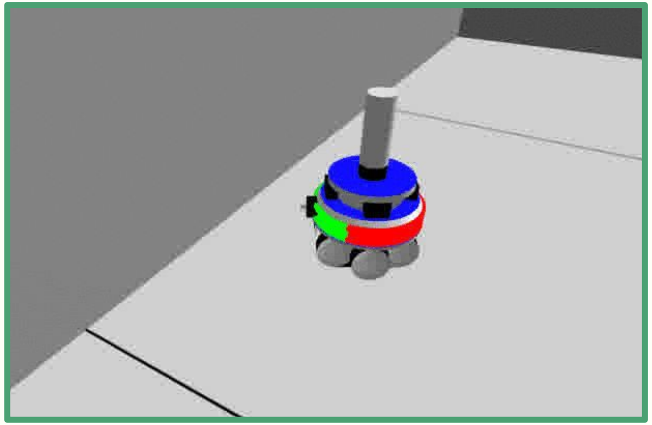
- Footbots
  - Swarmanoid Project
- 24 Proximity Sensors
- 24 Light sensors
- 12 LEDs in a ring
- LED Beacon
- Omnidirectional camera
  - Can detect different colors
  - Blob detection
  - View Distance = H*Tan(ϴ)



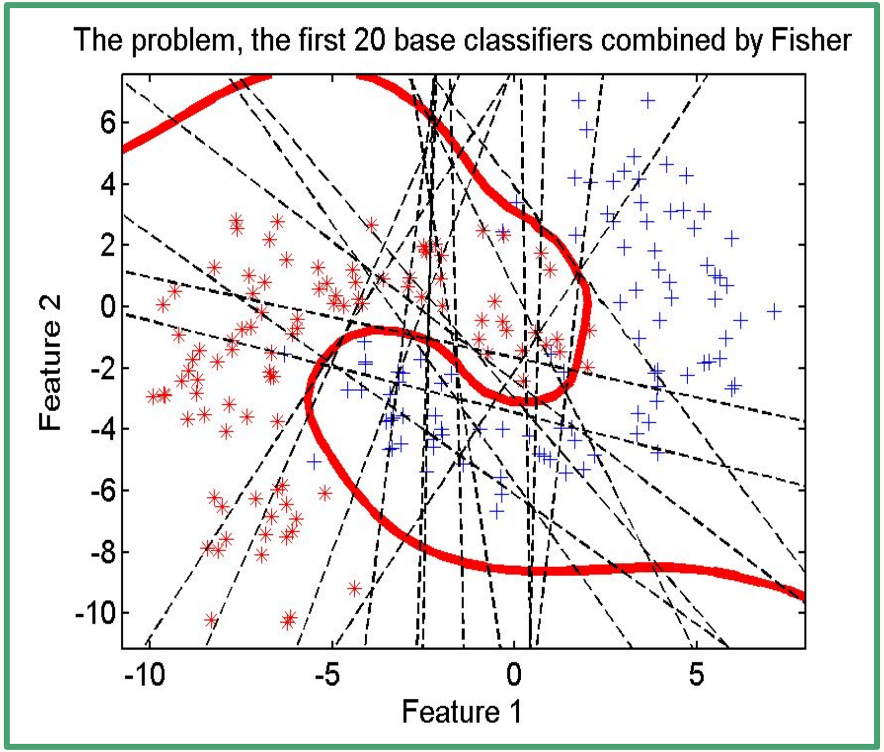
Methods and Theory

- LED ring is split into two along a virtual axis
- Robots are pushed by virtual forces
  - Attracted to Green and Repulsed from Red
  - Will rotate to direction of the net force
    - From blobs
    - From inertia
- Omni-directional camera detects blobs
- Blobs are used to calculate forces
  - Each force is inversely proportional to the distance the blob is away
  - Each force points in the direction of the blob

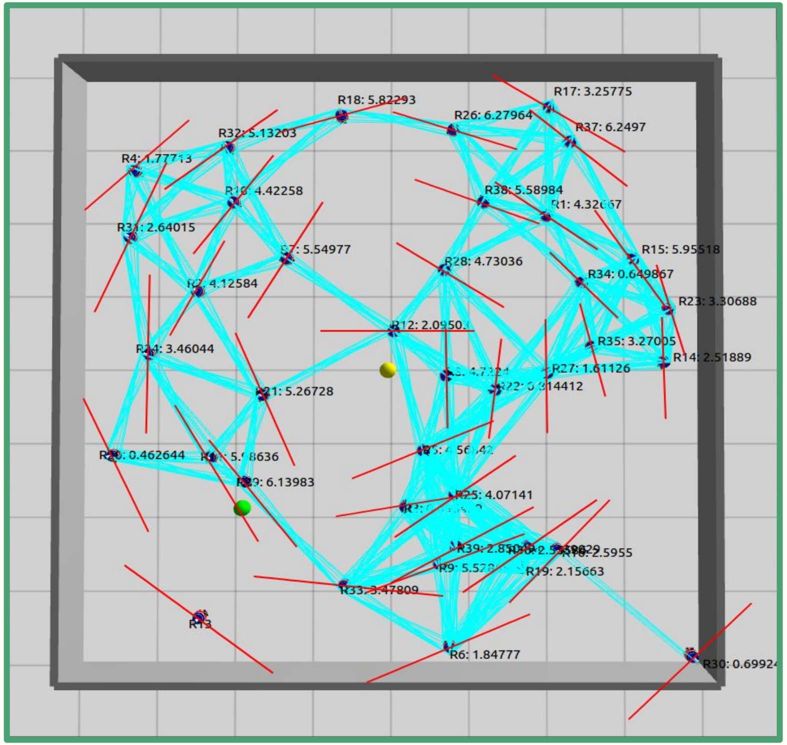




Methods and Theory

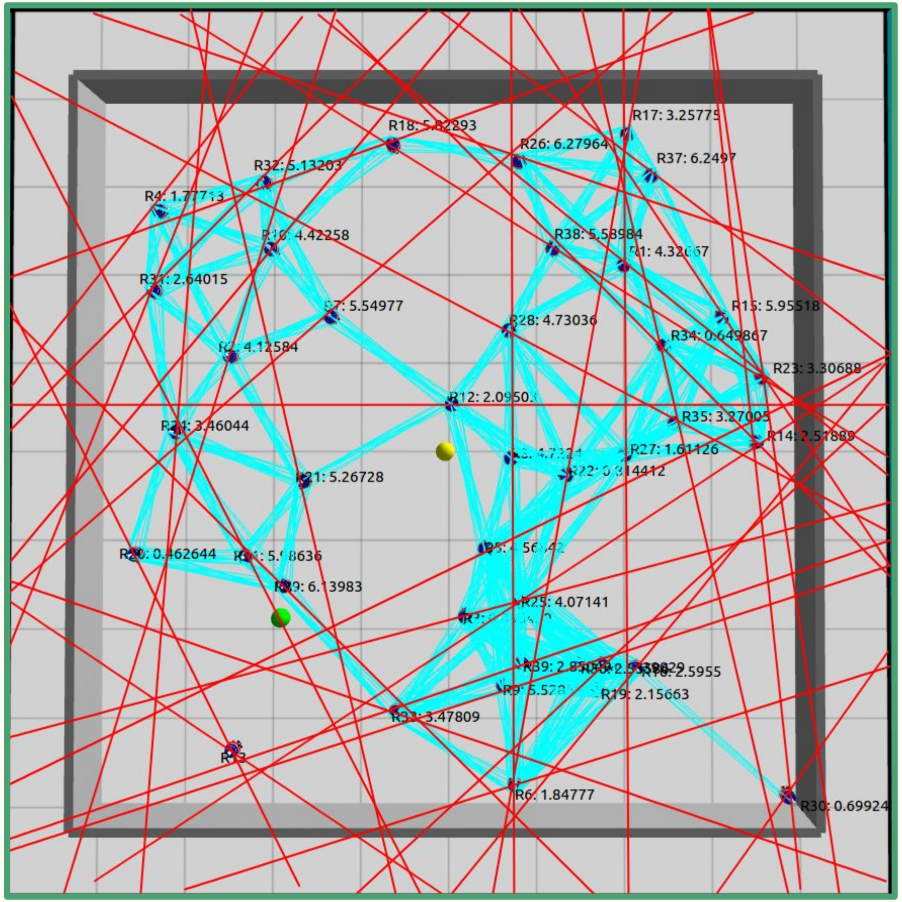


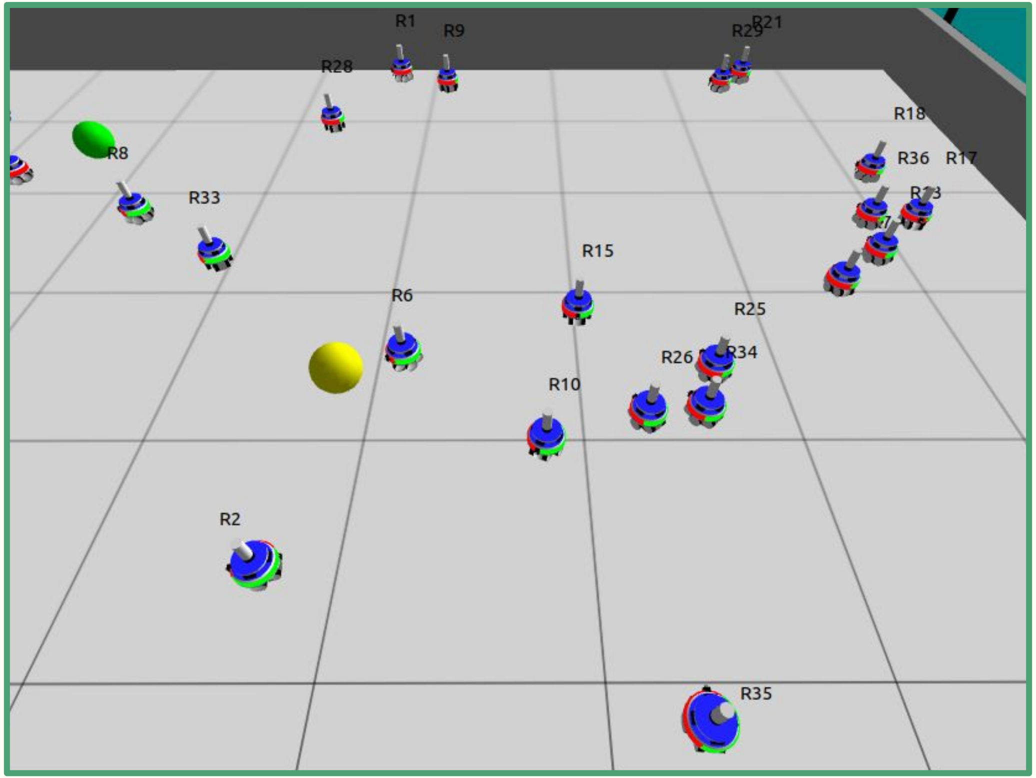
Ada Boosting



Our Model

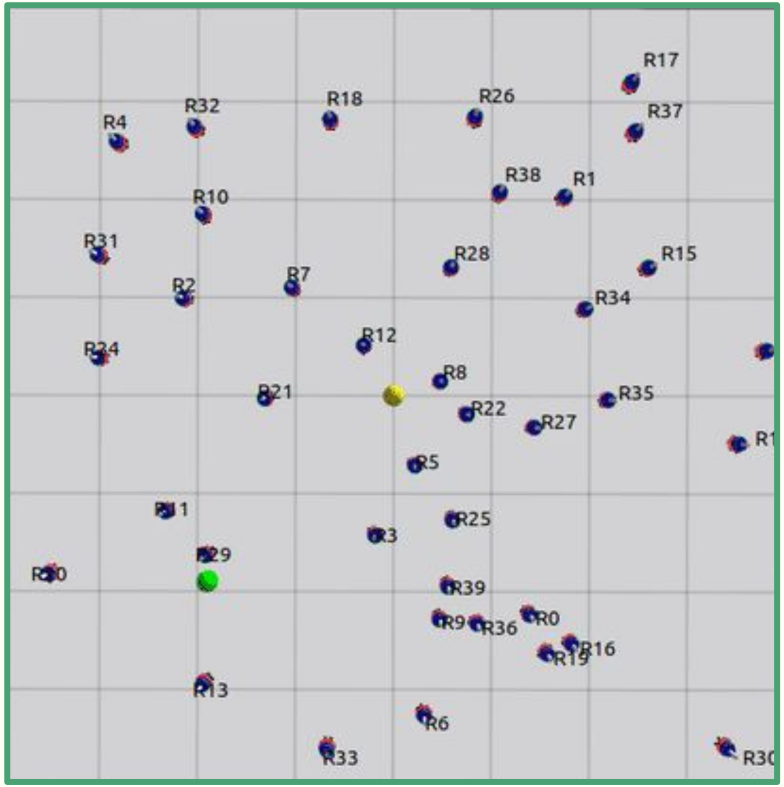
Methods and Theory



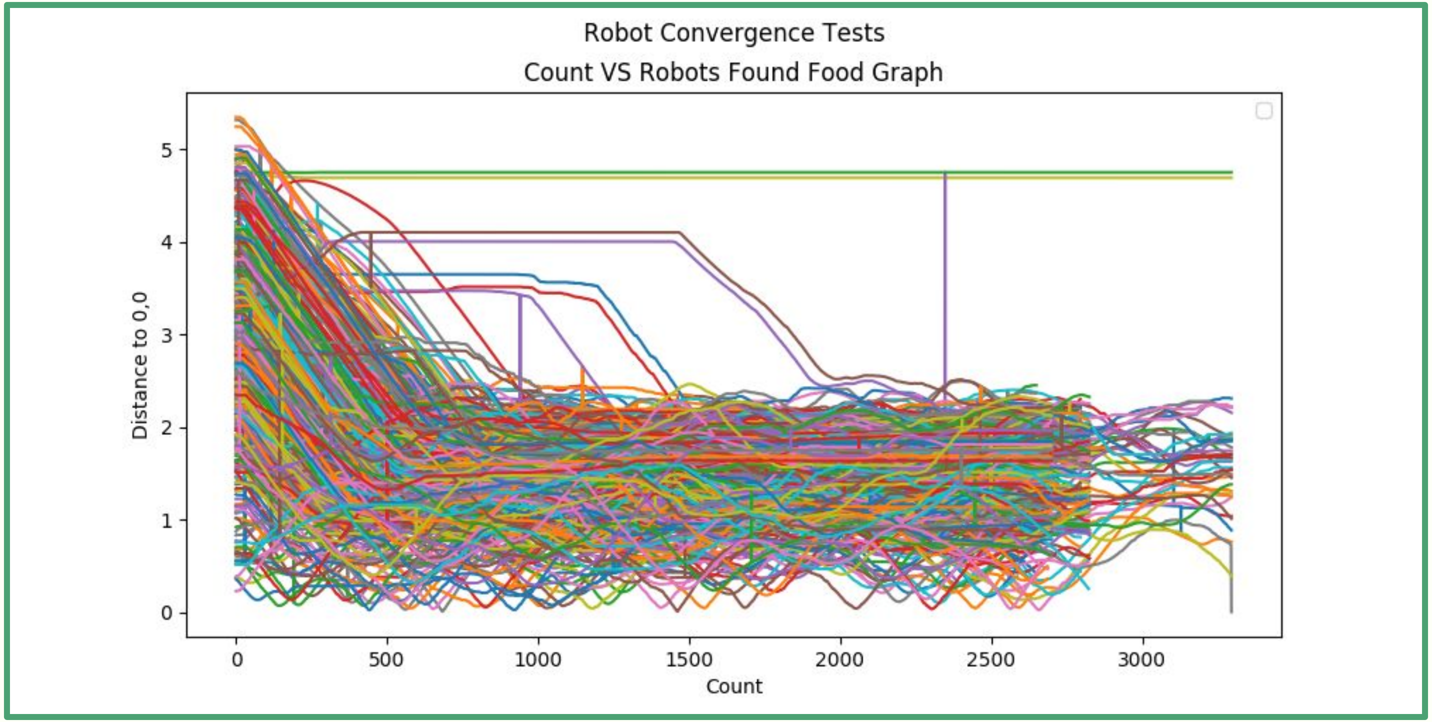

R1
R9
R29
R21
R28
R18
R36
R17
R8
R33
R15
R6
R25
R26
R34
R10
R2
R35

Convergence Testing

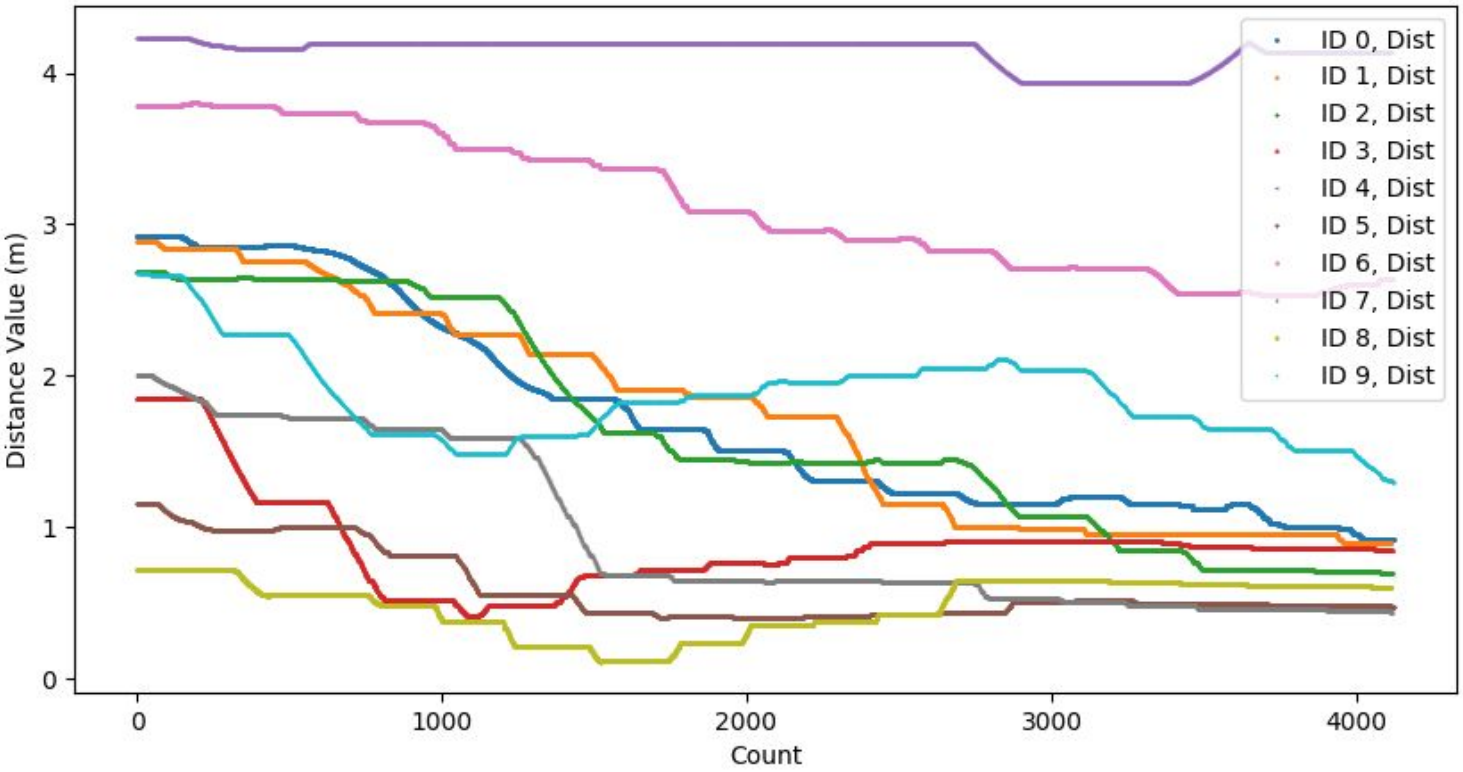
- All robots start with Known Point of Interest
- All robots start with virtual axis split
- All robots sum all force vectors created from all seen blobs
- All force vectors inversely proportional to the distance of the each blob
- All robots attempt to aggregate to given point.
- All robots rotate and move in direction of virtual force



Convergence Data



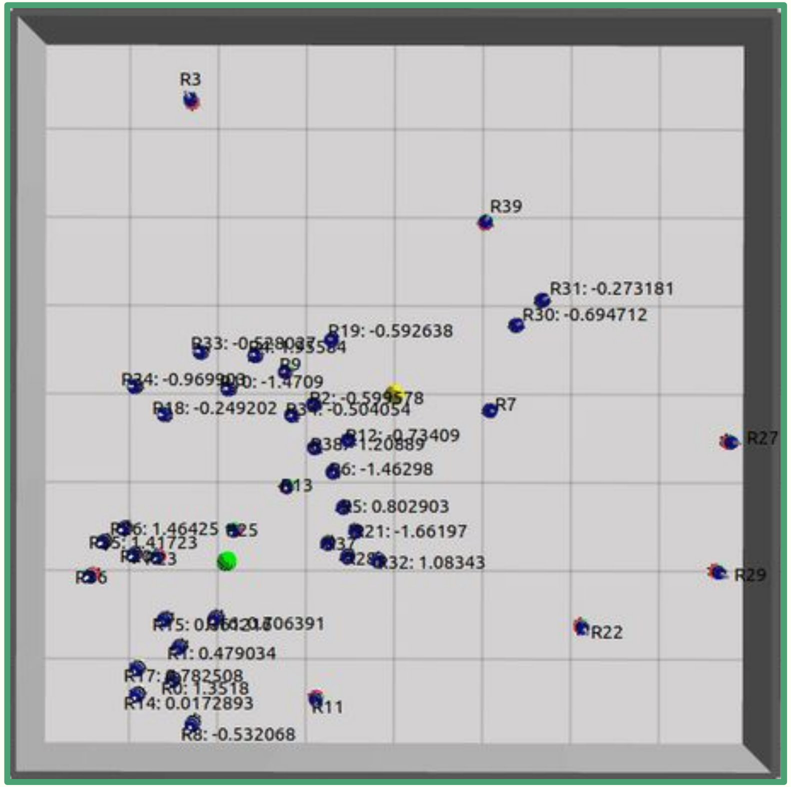
Convergence Trials



ID 0, Dist
ID 1, Dist
ID 2, Dist
ID 3, Dist
ID 4, Dist
ID 5, Dist
ID 6, Dist
ID 7, Dist
ID 8, Dist
ID 9, Dist
Distance Value (m)
Count
0
1
2
3
4
0
1000
2000
3000
4000

Food Finding Variations

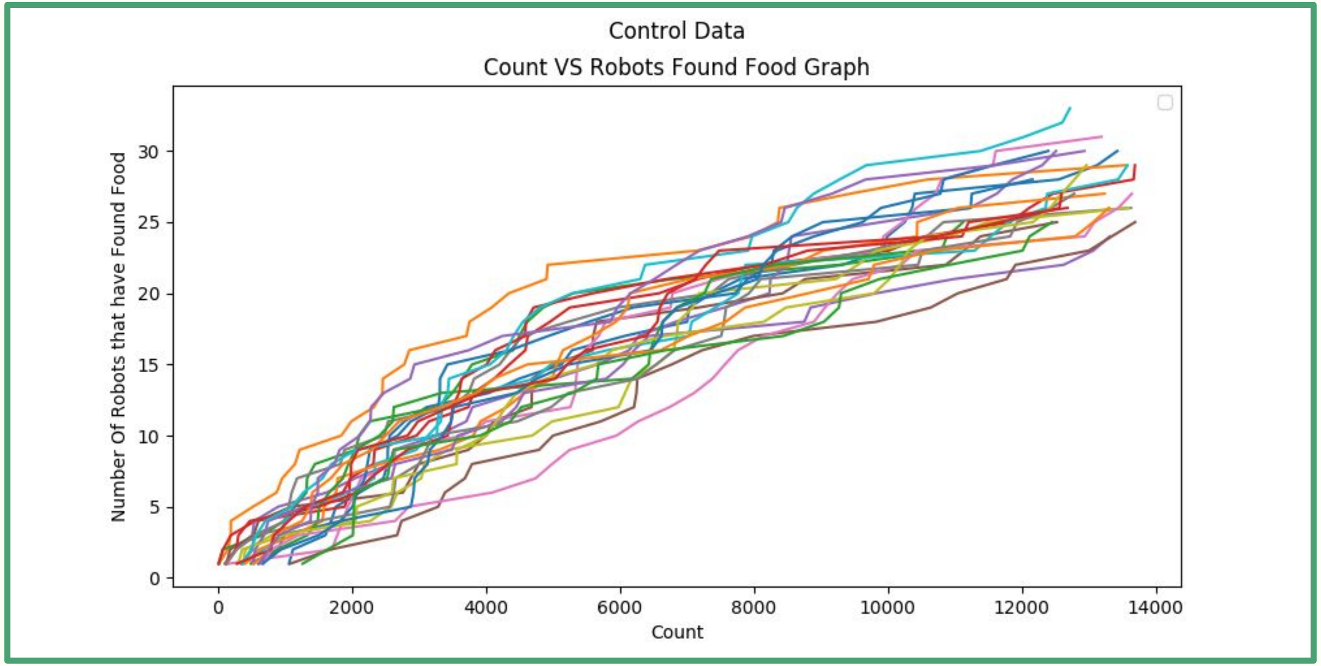
- Food Finding
  - No robots start knowing Point of Interest
  - If a robot finds the Point of Interest, they divide their region
  - Those that have not found Point of Interest, are steered by other robots' light sources.
  - Simulation ends when all robots have found the Point of Interest



Parameter Variations

Continuous vs. Discrete Turning	Does the robot stop when turning
Aperture Variations	How far the robots can see
Robot speed	How fast the robots travel
Turning speed	How fast the robots turn
Inertia factor	How much the robot is impacted by the blob detection forces
Repulsion and Attraction scale factor	-
Number of Robots	-

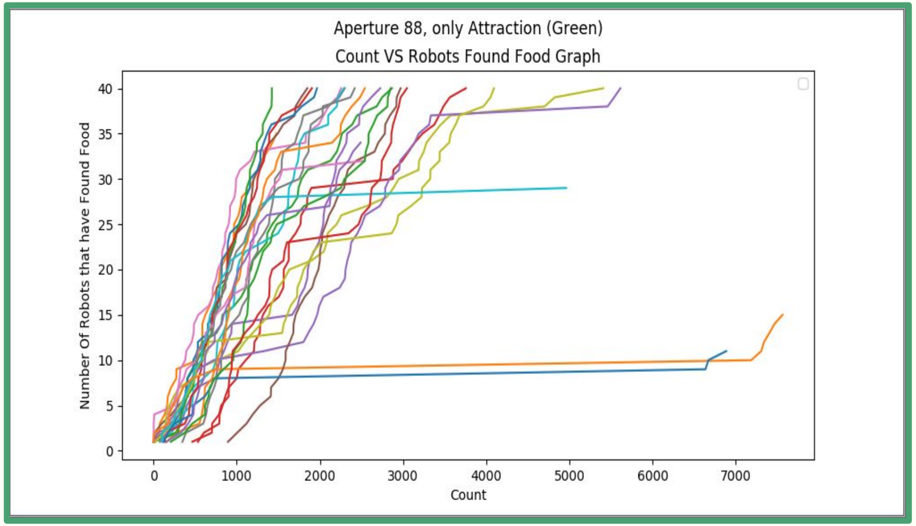
Food Finding Control Data



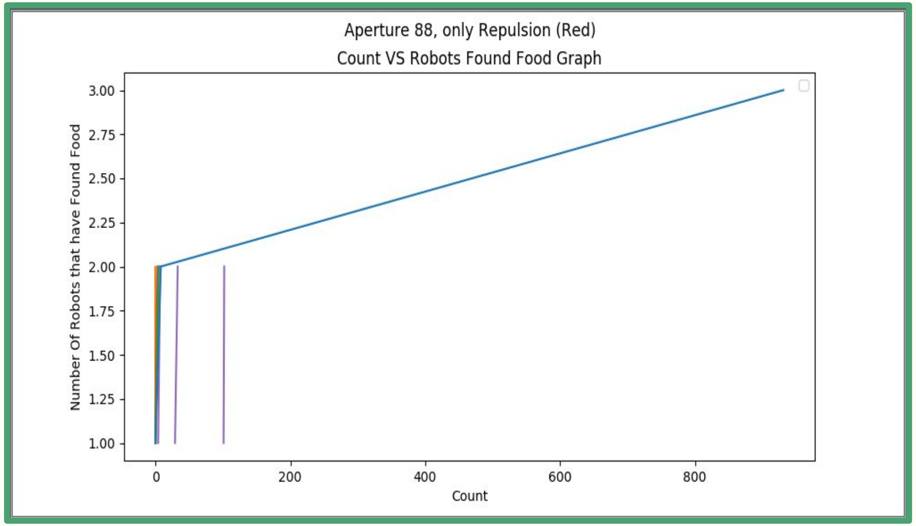
Control Trials

Food Finding Data

View Distance = 0.289*tan(88) = 8.276m



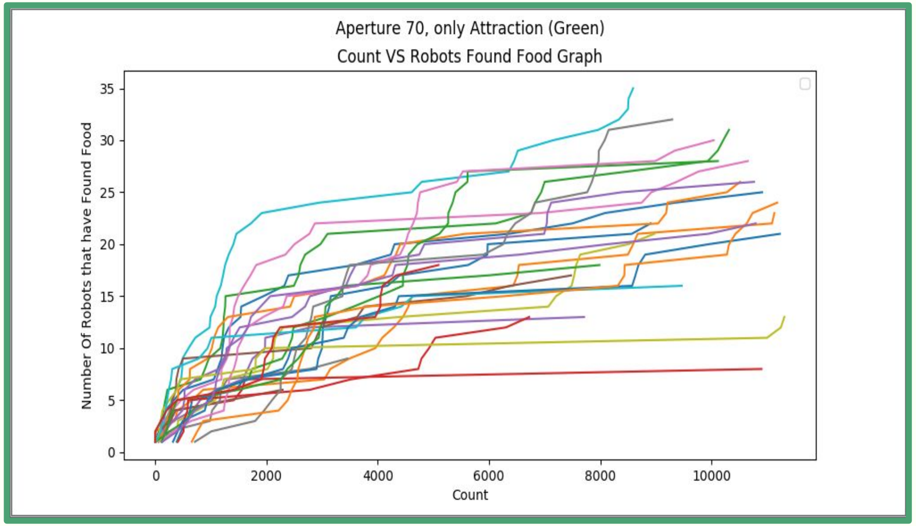
Green Only
Aperture 88

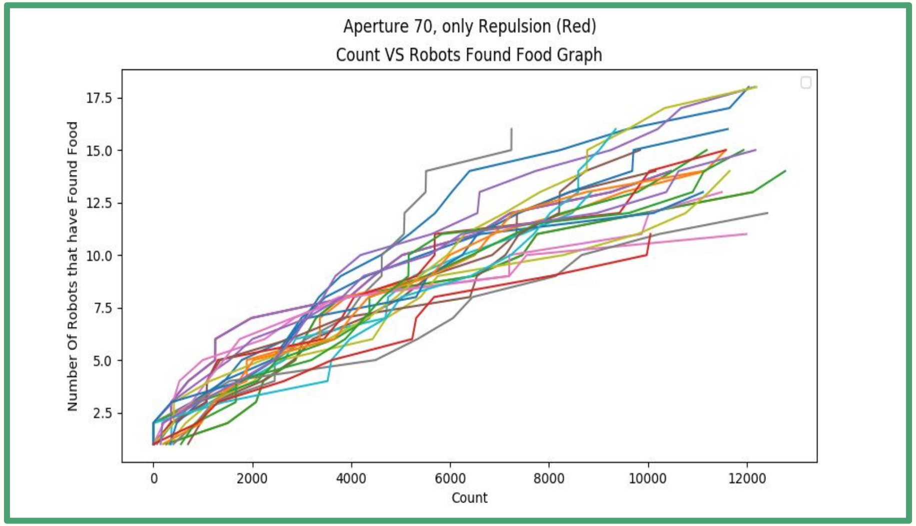


Red Only
Aperture 88

Food Finding Data

View Distance = 0.289*tan(70) = 0.794m

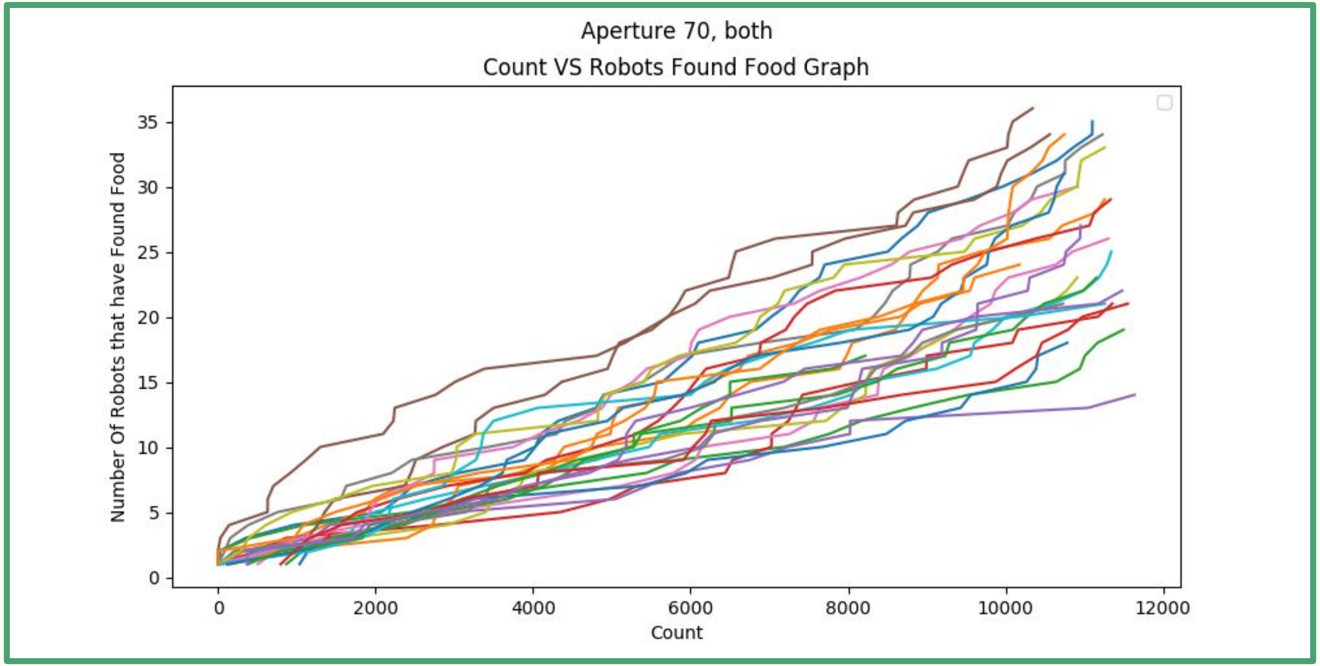




Green Only
Aperture 70

Red Only
Aperture 70

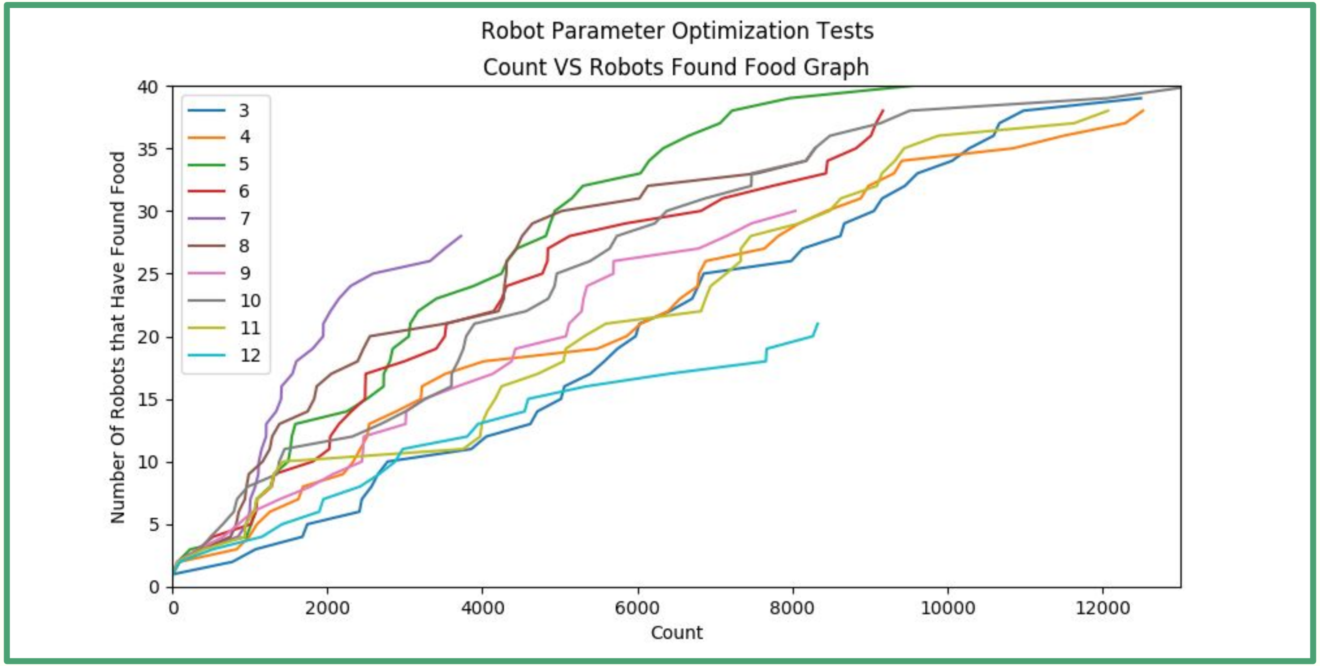
Food Finding Data



Both Red and Green: Aperture 70

View Distance = 0.289*tan(70) = 0.794m

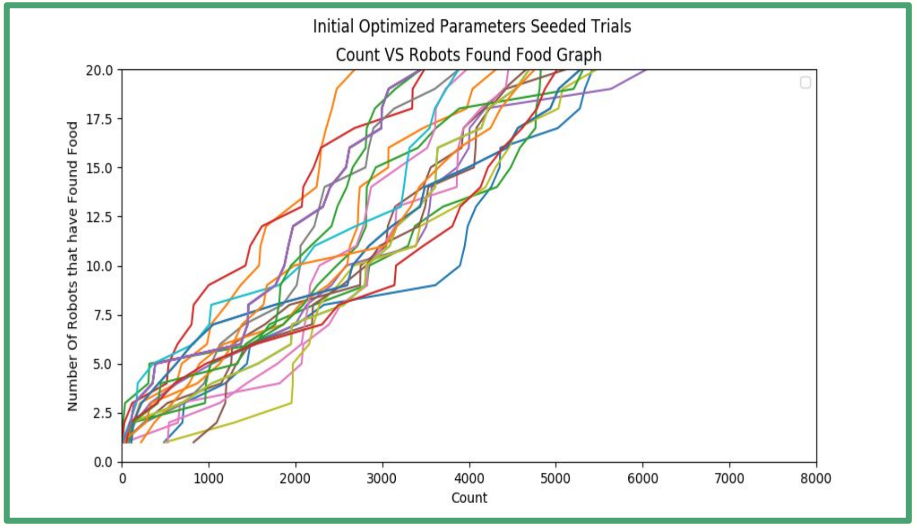
Food Finding Data



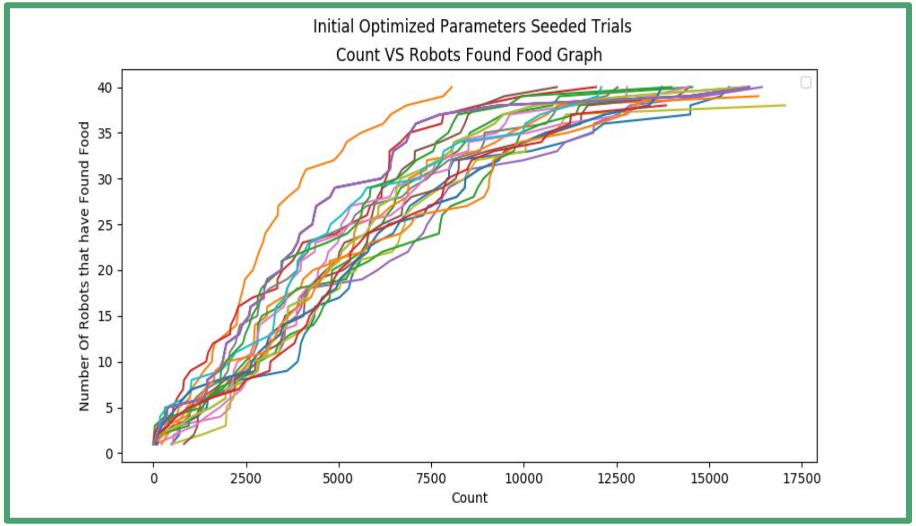
Optimization trials

Food Finding Data

View Distance = 0.289*tan(70) = 0.794m



50% Convergence



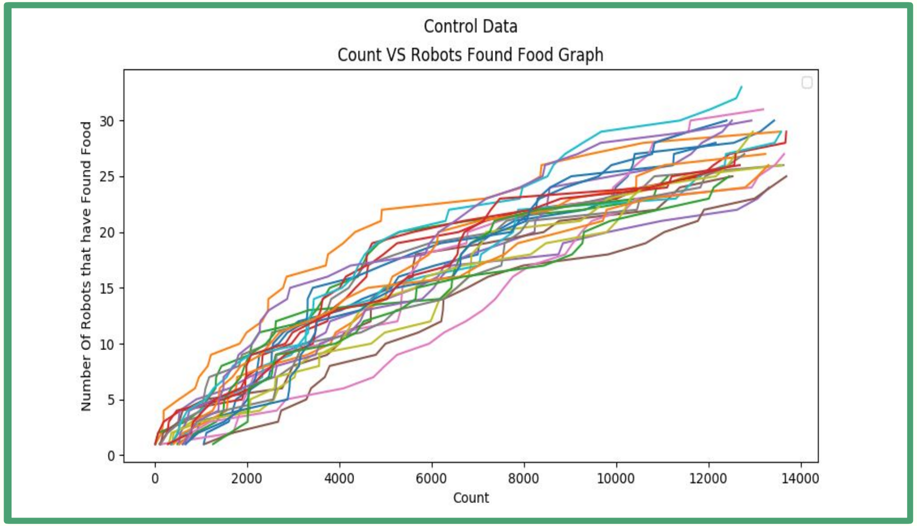
Full Trial

Optimized Trials

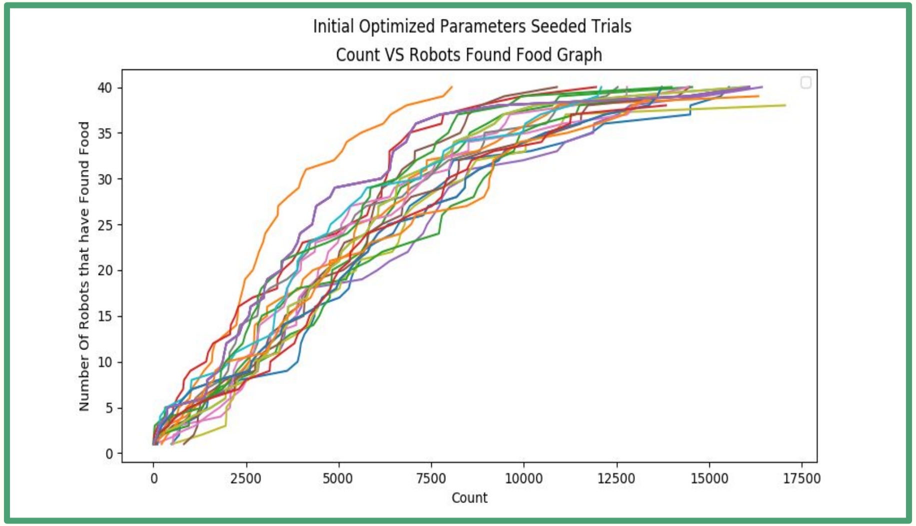
Model ≫ Control

Food Finding Data

View Distance = 0.289*tan(70) = 0.794m



Control Trials



Full Trial

Optimized Trials

Analysis

- Increased aperture size provided faster detection
  - Robots can see further
  - Ultimately not realistic
- Continuous turning is slightly more efficient
- Reducing repelling force factor improves detection
- Less robots decreases the efficiency of this model
- Other parameters can be optimized to improve the model and decrease time to detection
  - Inertia factor
  - Robot speed
  - Turning speed

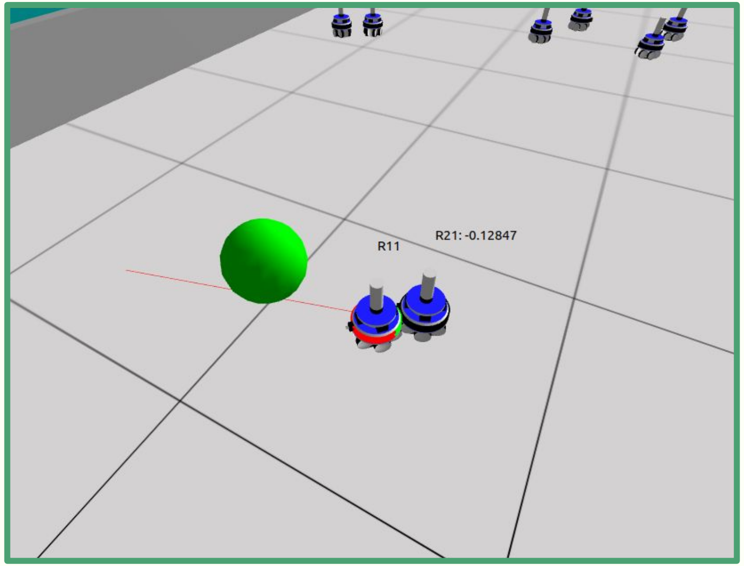
Analysis

- Attraction
    - Highly effective at long range, and when indicator robots are close to light
    - Misleading when indicator robots are far away from light
        - Robots will cluster around indicator robot and follow it around as it random walks
- Repulsion
    - Incredibly ineffective when there are few indicator robots at high range
        - Especially when close to light
    - As number of indicator robots increase, likelihood that their repulsion forces will cancel increases
        - Robots find food faster
- Both
    - At short range, will cause gravitational “hooking” towards POI
    - At high range will cause continuous influence towards POI

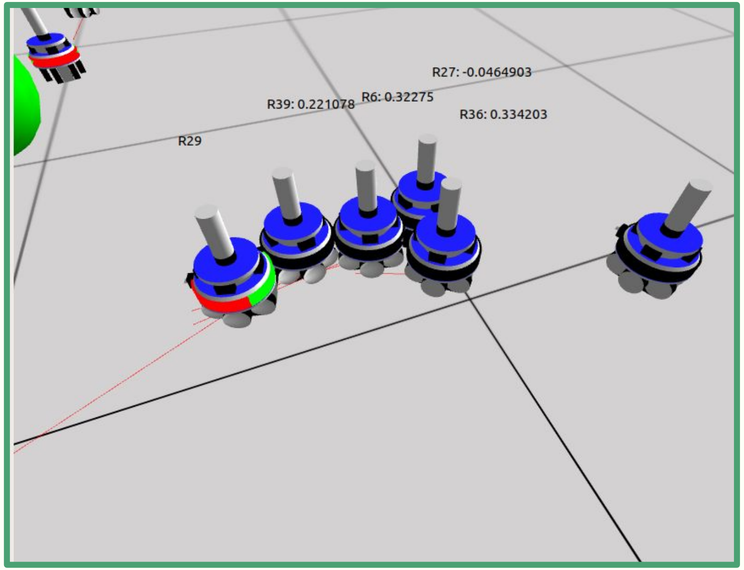
Adding Noise to the System

Separating Based On Region

Other Behaviors



Herding



Flocking

More behaviors?

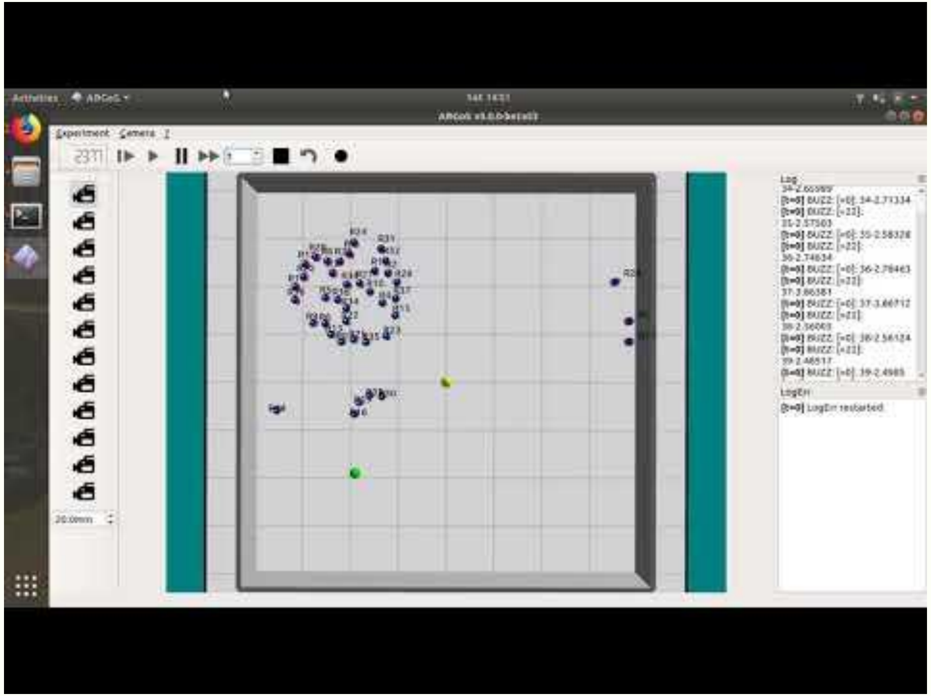
Closing Thoughts and Future Improvements

- In our simulations, the robots kept track of the Point of Interest
    - In real life, robots could keep track of the region they've divided using a gyro
    - No mapping needed
- Complex Separation regions allowing for zones of interest not just points of interest.
- Noise
    - Test robustness of model

Closing Thoughts and Future Improvements

- Increase number of LEDS
  - Higher resolution
- Increase number of colors
  - Allow for transfer of more directional information
- Variation in Random walks
  - Analyze their impact on the convergence time
- Use a different sensor to find a POI
  - Say a place with a temp that is too high, or a dangerous level of CO
- Using a Running Average of blob over multiple steps

Questions?



https://www.youtube.com/watch?v=0r2cxE1sSqo&list=PLXU3XMQtq5lSTmXgb7ol-OpTF3PWw1PFe

Bibliography

http://37steps.com/4993/random-fisher-combiner/

https://towardsdatascience.com/understanding-adaboost-2f94f22d5bfe