**Title**

This will be a presentation of my mid-project status for my project in:

Optimal UAV Base Station positioning for a temporary network by genetic algorithm

**Motivation**

Many different uses for UAVs: from pure enjoyment to military uses

Many UAVs working cohesively, or in a swarm, each equipped with a network antenna, could assist in search and rescue missions.

* means of communication for those searching
* particularly in remote areas with lacking existing communication infrastructure
* For example, if someone fell while hiking in a remote area of Alaska, this system could assist those searching for the hiker

Calculated positioning of the UAVs allows for more efficient coverage of users and could both keep costs lower and allow network coverage for more first-responders

**Background**

To calculate the height and coverage radius of the UAV, **Friis Transmission Equation** and **previous work from my lab** were used.

* Given specifications about the network module the UAV is equipped with, the equation can be used to calculate the maximum height at which acceptable network connectivity can be provided.
* Once this is solved for, the coverage radius can be solved for using simple trigonometry

**Genetic Algorithms** are also used in the solution.

* Fill a population data structure with random but appropriate guesses to a solution
  + “Appropriate guess” is defined as every guess has equal likelihood of being the correct answer
* Each guess is ranked by its **fitness** or extent to which it is correct
* **Natural selection** creates new generations by combining viable solutions from the previous generation
  + Solutions with higher fitness values are more likely to pass some information onto the next generation
* This process repeats until a certain fitness threshold is met - a solution has been found

**Methods**

**Calculations** previously described are performed first. The network module specifications are variable.

**Genetic Algorithms** - as previously described.

* The population is a list of lists containing 2-D coordinates
  + Each entry in the overall list is a list of UAV locations - a solution
  + These coordinates are UAV locations
* In performing natural selection, information from two previous solutions are combined to form one new one by taking the x-coordinate from one and the y-coordinate from another.
  + That’s how 2 parent solutions create a child solution

**Iterative Approach**

* The proposed genetic algorithm can only handle the location of UAVs as a variable, not the number of UAVs
* To combat this, an iterative approach is used:
  + The algorithm attempts to cover the area with one UAV
  + If it fails, two are tried, repeated until done
* This would be very computationally costly, but each failed run of the algorithm finds a way to cover as many users as possible
  + This is used for each subsequent run by filling some of the next initial population with the best solution from the previous GA run
  + For example, the algorithm initially finds the best placement for one UAV, and passes this information onto the next GA iteration with 2 UAVs

**Parameter Selection** - There are 4 main factors independent of the input data that impact the performance of the algorithm

* Population size - how many guesses should be maintained through all generations
* Mutation Rate - rate at which randomness should be introduced into the algorithm
  + Ensures the algorithm does not become fixated on one solution when there may be a better one
* Iteration Inheritance - how much of the population should be replaced by the best solution from the previous run as opposed to random guesses
* Generational calculation limit - how long the algorithm should try before adding another UAV and restarting

To find the best values for these 4 parameters, experimentation was done. The graph for population size is shown on the right.

* The population with the lowest runtime, while still finding an optimal solution for all trials was chosen
* In this example, 400 was chosen because 300 did not always find an optimal solution all 3x
* This process was repeated for all 4 parameters

**Validation Plan**

Observation of a wide variety of solutions were analyzed to ensure the outputs made logical sense.

**K-means clustering** will also be compared with.

Initial Observations:

* K-means is faster
* K-means does not account for the coverage radius of the UAV
  + (on right) red section probably could not be covered by one UAV
* K-means cannot account for multiple UAVs
  + # of clusters is taken as a parameter

**Preliminary Results**

Initial results are that:

* Proposed algorithm finds an optimal solution 70% of the time
* K-means clustering finds an optimal solution 25% of the time

When the above fail:

* the proposed algorithm presents a solution that is slightly less optimal than possible
* K-means clustering will yield a solution that will not cover every desired user
  + Due to the fact that k-means does not take coverage radius into account

Conclusions

* The proposed algorithm finds an optimal solution more frequently than k-means clustering
* The data from every failed run of the proposed algorithm is still usable, albeit less efficient
  + A kmeans fail will yield an incorrect solution to the problem

**Q/A**

Any questions?