Training Neural Networks with a Genetic Algorithm for Obstacle Avoidance in Simulated Autonomous Drones

Will Baumgartner

Drones

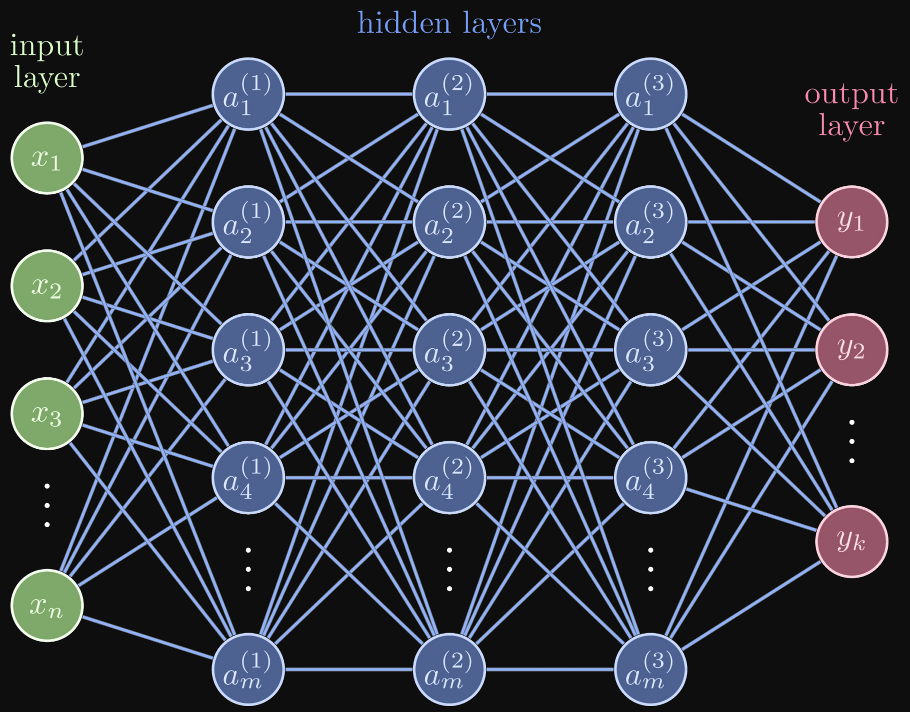
Unmanned Autonomous Vehicles (UAVs) or drones, for short, have become increasingly popular for many different modern uses. Drones can be used for anything from environment reconstruction through seed planting to any sort of aerial coverage.

A drone used for planting trees from above (DroneSeed)

A surveillance drone equipped with a camera (DJI)

Neural Networks

A **neural network** can be described as a function that attempts to transform an input to a given output through rigorous training and optimization, mimicking the way the human brain works. Deep neural networks are composed of an input layer, hidden layers, and an output layer. Each of these layers contains weights and biases, scalars that are slowly optimized to output the correct value for a given input, usually based on a given dataset.



A multilayer perceptron, represented as as connected nodes (TikZ.net)

Genetic Algorithms

A genetic algorithm (GA) is a method for solving optimization problems based on natural selection. In general, a GA works something like this:

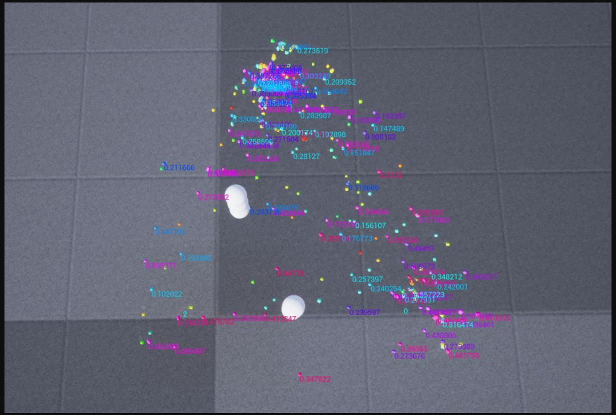
1. Initialize a population of individuals with random genes (initially performs badly)
2. Simulate population in environment and apply rewards and penalties for actions
3. Sort population by fitness, take a fraction of the best to be next generations parents
4. Cross-breed and mutate previous parents, replenishing the population
5. Continue generations until desired criteria is met, or individuals are sufficiently fit

Eventually, the best "genes" or traits from individuals will slowly accumulate, and individuals will reach a greater and greater fitness score. This is how a GA performs optimization.

Training Drones with a Genetic Algorithm

In this study, we want to train simulated drones to reach waypoints while avoiding obstacles along the way. Traditionally we could program an algorithm to change the path of the drone when it gets too close to an object, but this study aims to find whether or not a genetic algorithm is more viable for the task. Compared to traditional algorithms, a genetic algorithm can quickly find a nuanced and complex strategy for optimizing fitness.

In this study, simulated drones are equipped with a neural network controller which is optimized by a genetic algorithm with adjustable parameters, like population, species size, mutation rate, and percent eliminated.



All species of drones shown flying around course 2, with triangular waypoints and two obstacles.

Training Drones with a Genetic Algorithm (cont.)

To train our neural network equipped drones with a genetic algorithm, we need to specify some criteria for training. We want our drones to make it to waypoints with safety as their priority, and we can score fitness as follows. Drones with a higher score are more fit, and move on to the next generation.

Action	Score
Waypoint Reached	+1
Drone Collision	-4
Obstacle Collision	-40

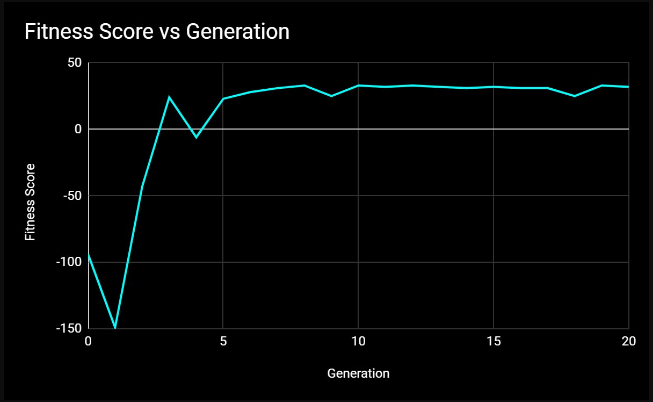
It's important to note why scoring values are of different magnitudes. We want to stress collisions more than reaching waypoints, because it's a much more costly maneuver. Also, it's much worse for a drone to collide with an object than another drone, so we prioritize that more. Also, scores aren't given per drone, rather per species, controlled by the same neural network, as we want the drones to learn on interactions with each other.

Why?

As drone technology becomes more viable for different applications, it's important to keep energy usage efficient and maintenance low. Even though drones are already very energy efficient, with increasing carbon emissions, its especially important to make sure we aren't using any extra energy that we could be saving. This project aims to make the safest drone possible that makes it to its destination in the least amount of time.

Results

This is the data from three different looped waypoint courses comparing reached waypoint counts, drone crashes, obstacle crashes, and fitness for drones trained with traditional and genetic algorithms. Parameters are kept **constant*** throughout.



We can see in an example run that the genetic algorithm is learning over each generation, eventually peaking!

Top Performing Generations

	Course 1 (safe)	Course 1 (fast)	Course 1 (GA)	Course 2 (safe)	Course 2 (fast)	Course 2 (GA)	Course 3 (safe)	Course 3 (fast)	Course 3 (GA)
Fitness Score	33	-20	37	33	-7	41	22	15	36
Waypoints Reached	33	41	37	33	41	41	22	31	36
Drone Crashes	0	8	0	0	2	0	0	4	0
Obstacle Crashes	0	1	0	0	1	0	0	0	0

The above table shows that the GA is clearly better than both safe and fast traditional algorithms. It's interesting to see that while the fast algorithm gets to more waypoints, it sacrifices more collisions!

* 32 species, 8 drones per species, 2>8>16>16>8>1 layer MLPs, 50 generations, 20 seconds per generation, same courses, best performing neural networks shown

Conclusion

In conclusion, it is clear that the genetic algorithm was able to perform better than the traditional algorithm, even though both given the same amount of data. The traditional algorithm could have been further improved, but the genetic algorithm showed that it can quickly and accurately solve a problem, without the cumbersome trial and error of designing a traditional algorithm. Machine learning is applicable to so many areas, and it proves itself viable in many cases.