Find Best chromosomes of Goats

INFO6205_201
Final Project

Team Member:

Yumeng Chen (001409547)

Dezhi Yang (001447703)

1. Introduction

Genetic Algorithm

A genetic algorithm (GA) is a search algorithm used in computational mathematics to solve optimization problems. It is a type of evolutionary algorithm. A genetic algorithm is usually implemented as a computer simulation. For an optimization problem, a certain number of candidate solutions (called individuals) can be abstractly represented as chromosomes, allowing the population to evolve towards a better solution. Traditionally, the solution is represented in binary (a string of 0's and 1's), but other representations can also be used. Evolution starts with a population of completely random individuals, and it happens from generation to generation. Evaluate the fitness of the entire population in each generation, randomly select multiple individuals from the current population (based on their fitness) and generate new life populations through natural selection and mutation. This population becomes the current population in the next iteration of the algorithm.

Background

Northern goats are also called hanging sheep, wild goats, etc., with a body length of 105-150 cm, a tail length of 12-15 cm, a shoulder height of about 100 cm, and a weight of 40-60 kg, but the maximum weight can reach 120 kg.
The ibex inhabits the bare rocky plateau and mountainside rubble ridges at an altitude of 3500-6000 meters. It does not migrate to low places in winter, so it can be called one of the mammals with the highest habitat. Very good at climbing and jumping, the hoof is extremely solid, the elastic sacroiliac joints and toes like pliers. Distributed in northern India, Afghanistan and Mongolia, in China in Xinjiang and northwestern Gansu, northwestern Inner Mongolia and other places.
In Biezhentao Mountain in Wenquan County, Xinjiang, China, after entering August, the northern goats began herding after going to the winter rams and ewe grouping. After the mating was completed at the end of October, they regrouped again. When the people congregate in August, the rams will live from their own living territory to the ewe's territory for a period of time. This process is the process of migration.

Idea[i]

Genetic algorithms are well-suited to problems with huge solution spaces and are useful for problems that have no analytical solution. In this project, we want to find genes that is the best in the reproduction of the goats.

2. Algorithm Design

2.1 Expression Algorithm[ii]

Overview

Expression refers to the mapping between genotypes and phenotypes. By analyzing the genotype, we can get the individual's route in the maze to get the phenotype. A genotype consists of 128 genes, which have three modes of action - forward, clockwise rotation and counterclockwise rotation.

Working flow

First, do the following for a "chromosome" array of length 128:
We treat every two genes as a group. If the first number is 1, then +1. If the second number is 1, then +2. For example: [0,0] means 0; (1,0) means 1; [0,1] means 2; [1] means 3. Now we get an array of 64 bits in length, represented by 0, 1, 2 or 3.
Then, make the following judgments on the elements in the 64-bit array:
If it is 0, this person will not move; if the individual does not reach the exit and does not complete the maximum number of moves, the following judgment will continue:
If it is 1, this person will take a step forward. The direction of motion depends on the direction it is facing. If it is facing the wall or beyond the scope of the map, it cannot continue to move.
If it is 2, the object rotates 90 degrees clockwise and does not move forward.
If it is 3, the object is rotated 90 degrees counterclockwise and does not move forward.
In addition, which value to choose in a 64-bit array depends on the perception of the surrounding environment.
If there is a wall in front, +1; if there is a wall on the left front, +2; if there is a wall on the right, +4; if there is a wall on the left, +8; if there is a wall on the right, +16; if there is a wall behind, + 32. This will give you an integer [0,63].

Conclusion

Therefore, we successfully mapped the genetic map of the mobile route. Two key points are that the algorithm is highly fault tolerant. In the 128-digit gene sequence, if some sites are mutated, it does not necessarily lead to phenotypic changes; unlike "random movement", only the two types of "genotype" and "environment" and "gene expression" are the most match.

2.2 Elimination Algorithm[iii]

Overview

For individuals, elimination is based on three aspects:

By accident, young or with high fitness individuals may be eliminated;
Individuals who are too old may be eliminated;
Individuals with low fitness may be eliminated;
When selecting the individuals to be eliminated, we make the following judgment: first traverse the entire population, get the individual set of "age" greater than 10 years old, and put it at the end of the entire group list; according to the fitness, sort the remaining individuals in descending order. Place these young individuals in front of older individuals and divide the entire list into two parts of the same length. Finally, the probability of death in the first part is set to 20%, and the probability of death in the second part is 80%.

Purposes

This can achieve the following purposes:
Older individuals and individuals with poor constitution are more likely to be eliminated, which is conducive to the evolution and development of the population;
Since we don't know the genetic relationship between new individuals and parents, we need to ensure that new individuals can compete fairly with their parents. If the gene of the newborn individual is superior to the parent's gene, we can guarantee to some extent that these excellent genes are preserved, so that the entire population can be evolved.

2.3 Reproduce Algorithm

Purposes

Hybridization is the process of individual reproduction. Observe the entire population and perform the following operations for each individual.

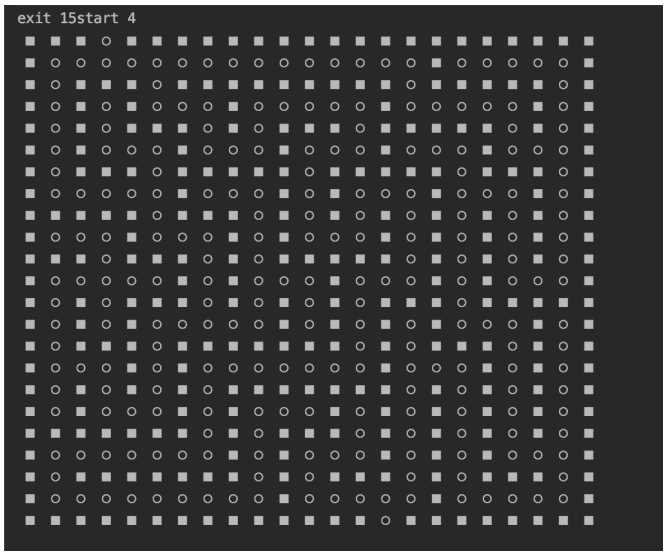
Working Flow

Select best top 10(tournament size) fitness individuals which age are below 10 as parents, in this situation we will ignore the sex.
Determine whether this individual is dead. If it dies, a new individual is created to replace it. Select parents randomly according to the from array above.
New individual should get 50% chromosome (left chromosome) from father and left 50% chromosome (right chromosome) from mother. Thus, generate a random number, when the number is less than 0.5, it gets left chromosome from father and right from mother. The number is greater than 0.5, it gets left chromosome from mother and right from father.
Mutation happens during this phase.Set mutation probability to 0.1. when randomly generated number is less than 0.1, if chromosome value is 1, set it to 0. If it is 0, set it to 1.
Calculate fitness rate of each individuals

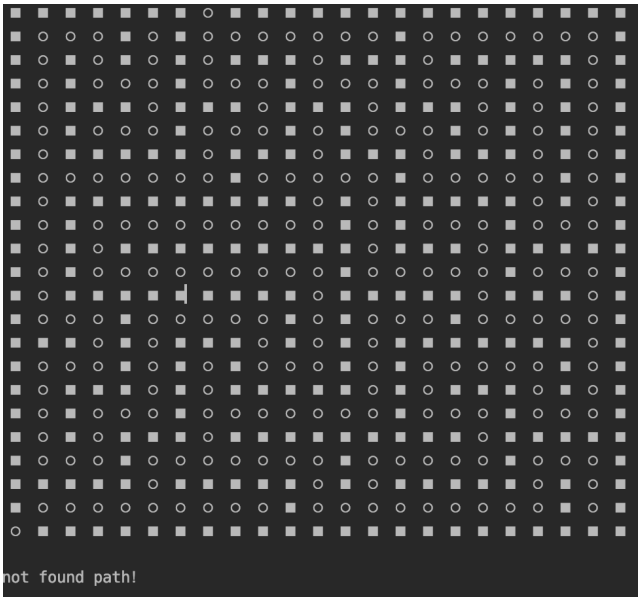
2.4 Maze Algorithm:

Purpose

Random generate maze which will be used to calculate fitness. Set random start point on first row in array and exit point(goal) in last row. And we only have one correct path in our maze.
When the maze generated not reach requirements, we will loop generate it until it qualified.



This is the qualified maze



This is not the qualified maze 1

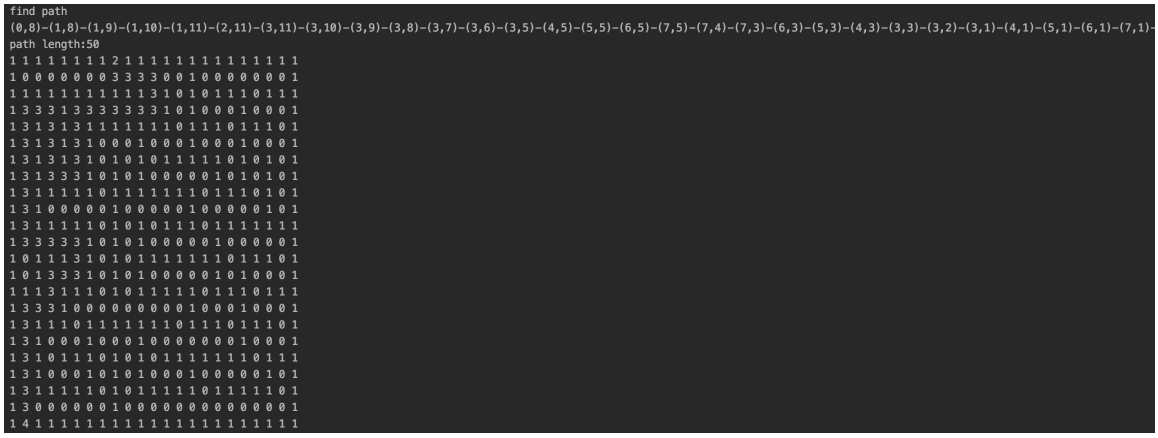
We can see that this maze cloud not find a path from start to exit.

Working flow

First, we call make Maze method in Maze class to initialize the start and exit in our array and set others all as wall (value 1). then call generatePath method inside makeMaze to

generate path include correct path in our maze. In generatePath method we define two – dimension array call dir which contains {{0,1}, {1,0}, {0,-1},{-1,0}} represent 8 different direction. Variable turn assists with next decide out next path may generate.
After we get maze as above shows, then we using dfs algorithm to find unique correct path from start to exit.
Now we have get path in our maze as string, then we just need extract each node in our path using regex and set it as different value.

Here we set wall as 1, start point as 2, correct path as 3, exit point as 4, path can walk but not correct (not lead to exit) is 0. Now we have prepared the maze that we will use in genetic algorithm.



Conclusion

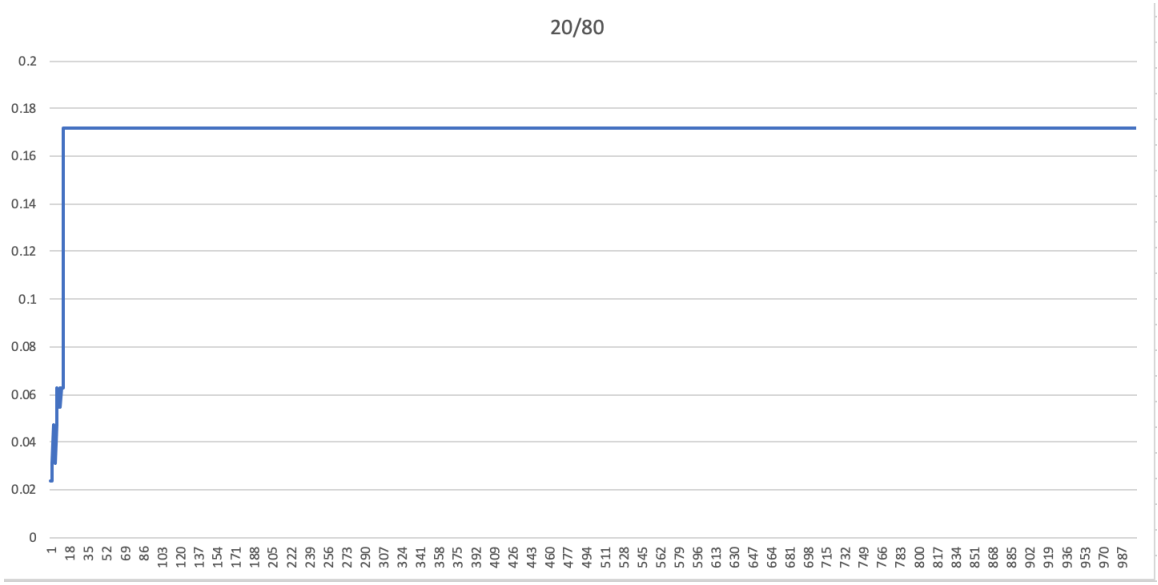
Then we need to calculate fitness of each individuals, after each individual finish their walk, they will get a total fitness. Only individuals walk through path which is marked as 3 will get 1 point. And divided by path length can get final fitness rate of itself.

Data Analysis

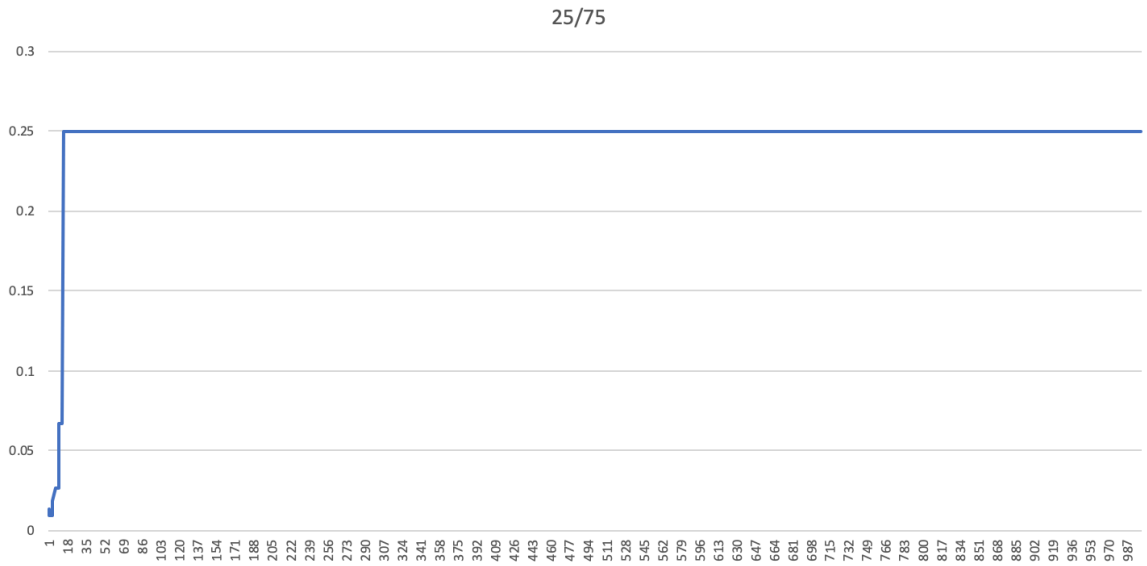
In the genetic algorithm, our initial population value is 1000 and the initial generation value is 1000, and we rank all individuals in each generation, placing individuals with high adaptability in the first half of the population, and individuals with low or older adaptability in the second half of the population.

i. 20% & 80%

In the first case, we set the death probability in the upper half of the population to 20% and the death probability in the lower half to 80%. We get results as shown below.

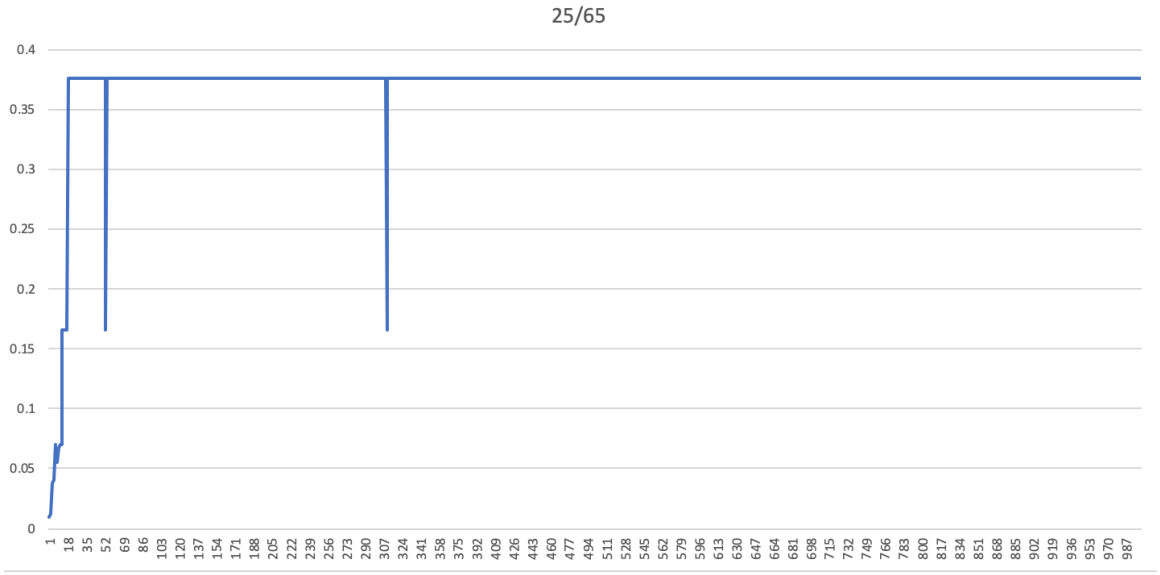


ii. 25% & 75%

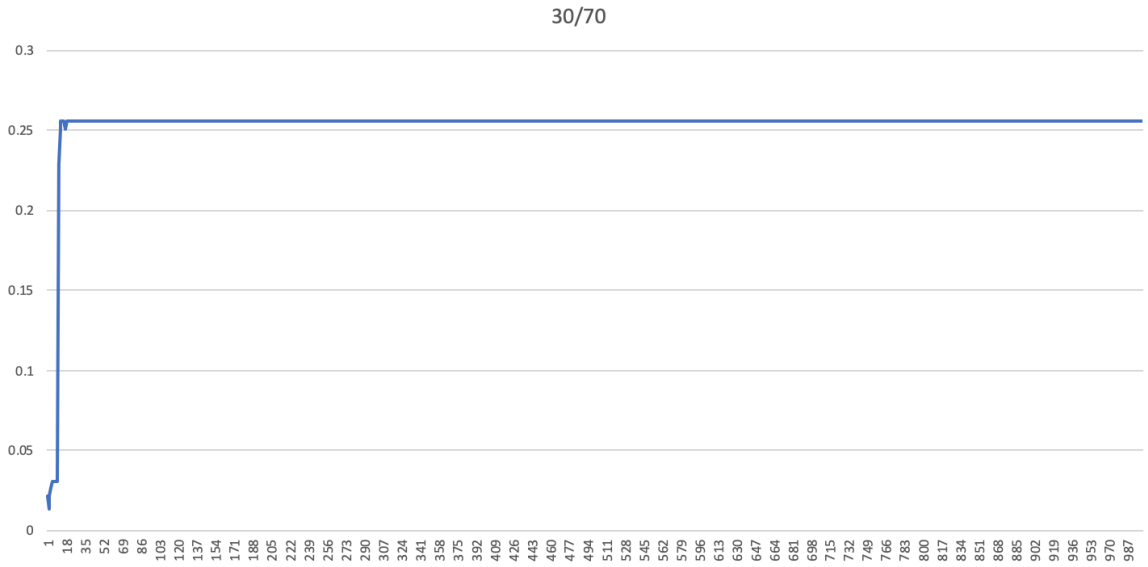


iii. 25% & 65%

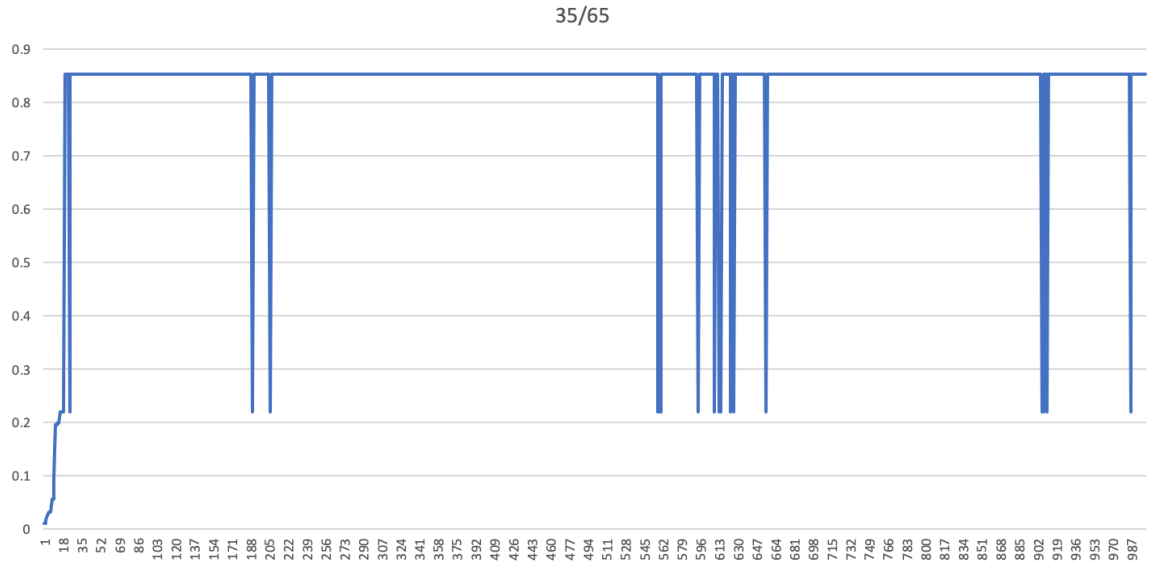
It's not required to make the sum of two probabilities equals to 1. So, this time, we continue set the death probability in the upper half of the population into 25% , where the death probability in the lower half to 65%, to check whether a lower dead probability of the lower half will make the evaluation more reliable.



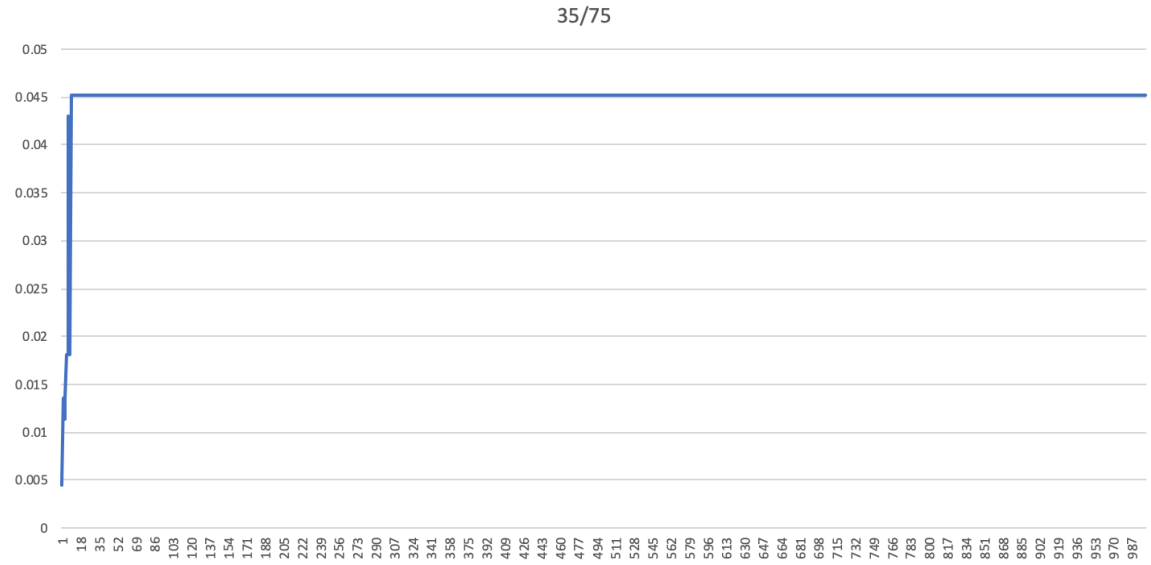
iv. 30% & 70%



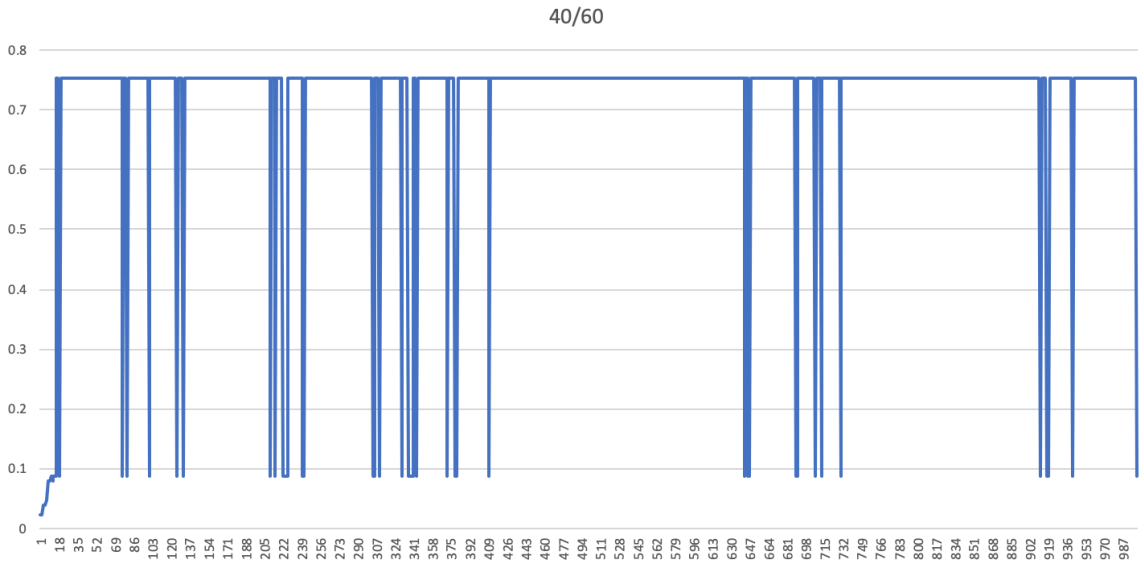
v. 35% & 65%



vi. 35% & 75%



vii. 40% & 60%



We do a lot of attempts of this pair of probabilities. We found that the population reached the limit of the evolution less than 50th which is very quick in the pictures above. We consider that this will not match the situation in the real world.

Conclusion

It shows roughly the same trend and will get healthy and stable genomes at a very fast speed. This may be because the maze is generated randomly, although we have made specific modifications to the number of steps in the calculation of the individual genome according to the generated maze, instead of setting a fixed value. However, it seems that the specific reasons that influence the results obtained are still caused by random maze. We should add more misleading paths to the maze to prevent individuals from evolving too quickly and when calculating fitness, we should also increase the corresponding health for individuals who have evolved on the wrong path. But we should still breed by selecting individuals who have a higher degree of evolution on the right path. This method can effectively delay the generation of stable genomes.

[i] Team 520, Zheng Wenkai, (2019), *Zebra Migration*, INFO6205, 2
[ii] Team 520, Zheng Wenkai, (2019), *Zebra Migration*, INFO6205, 6
[iii] Team 520, Zheng Wenkai, (2019), *Zebra Migration*, INFO6205, 7