

EquationSolver

Benjamin Katz

February 2022

1 Design Overview

I create a population of solutions with random genes(x and y values). Within a for loop for the max amount of generations, I check to see if any of the solutions are above the threshold, if they are not, I store the best solution and create a new generation. In order to create a new generation, I either compare each solutions fitness to a different solution and add the winner to the mating pool(Tournament). Or I allow the fitness as a proportion of the total fitness of the entire generation to determine how many of one specific solution to add to the mating pool(Roulette). Once in the mating pool, the portion that are designated to cross over do so and all others remain unchanged. The percentage of genes that are to be mutated are replaced with a random value. This continues until max generations or until a fitness above the threshold is found.

2 Fitness Criteria

Fitness is a double = $6x - x^2 + 4y - y^2$

3 Threshold

The threshold is given by the user, in my testing I used 13.

4 Selection Type

With a population size of 100, a mutation rate of .2, and a crossover rate of .5 roulette performs better than tournament around 60 percent of the time.

5 Mutation

If a random number is less than the Mutation rate, a given gene is swapped for a random number between 0-100. This is done for every gene during the mating process.

6 Crossover

If a random number is less than the Crossover rate, two new children are created with the x gene of one parent and the y gene of the other. These two children contain the genetic code of their parents but are distinct from each other.