Project 2: "Genetic Algorithms in LEAP"

Sai Thatigotla

COSC 420

2/25/22

Introduction

In this project we used an implementation of a basic genetic algorithm from the LEAP library and analyzed the effects of various parameters/arguments on the performance of the algorithm through various metrics.

Theory

The main idea of this project was to analyze how changes to the parameters and nature of the genetic algorithm (GA) affected its efficacy. GAs are algorithms that can be used to solve optimization problems and are inspired by natural selection and thus model many of its features and characteristics. We looked at the values given such as population size (N), mutation probability (p_m), uniform crossover probability (p_c), tournament size for tournament selection as well as calculated values such as average fitness for generation, the best fitness for generation, whether a solution (where fitness = 1) was found in the generation, how many solutions were found in the generation, and a measure of diversity of the population for each generation.

Methods

The default code provided was used to generate the data files. calc.py was created to calculate some more values and create results_v2.csv which summarizes all the results. graph.ipynb was used to create the plots. Every experiment was run for 30 generations and the parameter space tested was the following:

Population size = [25, 50, 75, 100]
Mutation probability = [0, 0.01, 0.03, 0.05]
Crossover probability = [0, 0.1, 0.3, 0.5]
Tournament Size = [2, 3, 4, 5]

The default parameters are N= 50, p_m = 0.01,p_c= 0.3, and tournament size at 2. Each combination of parameters was run for 20 iterations.

Results

Default Parameters

The following figures show the average fitness, the best fitness, and the diversity of each iteration for the default parameters.

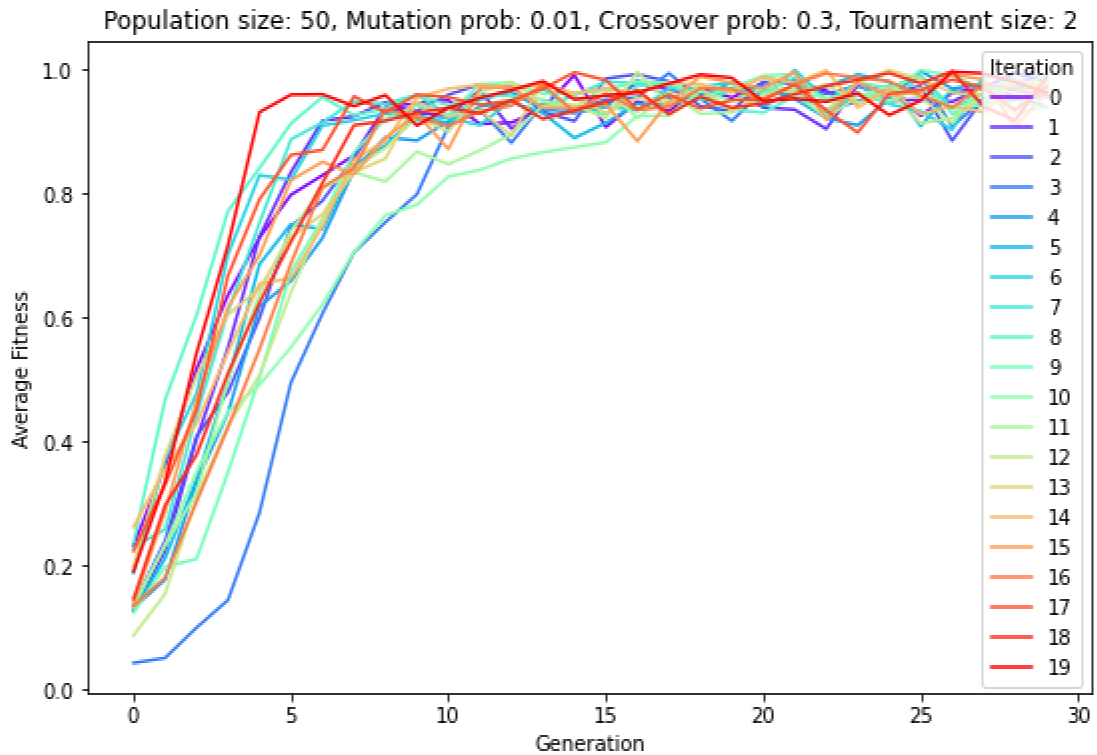

Figure 1. Average fitness

In general, for the default parameters, the algorithm converges within 10-15 generations at a fitness level of 0.9-1.0.

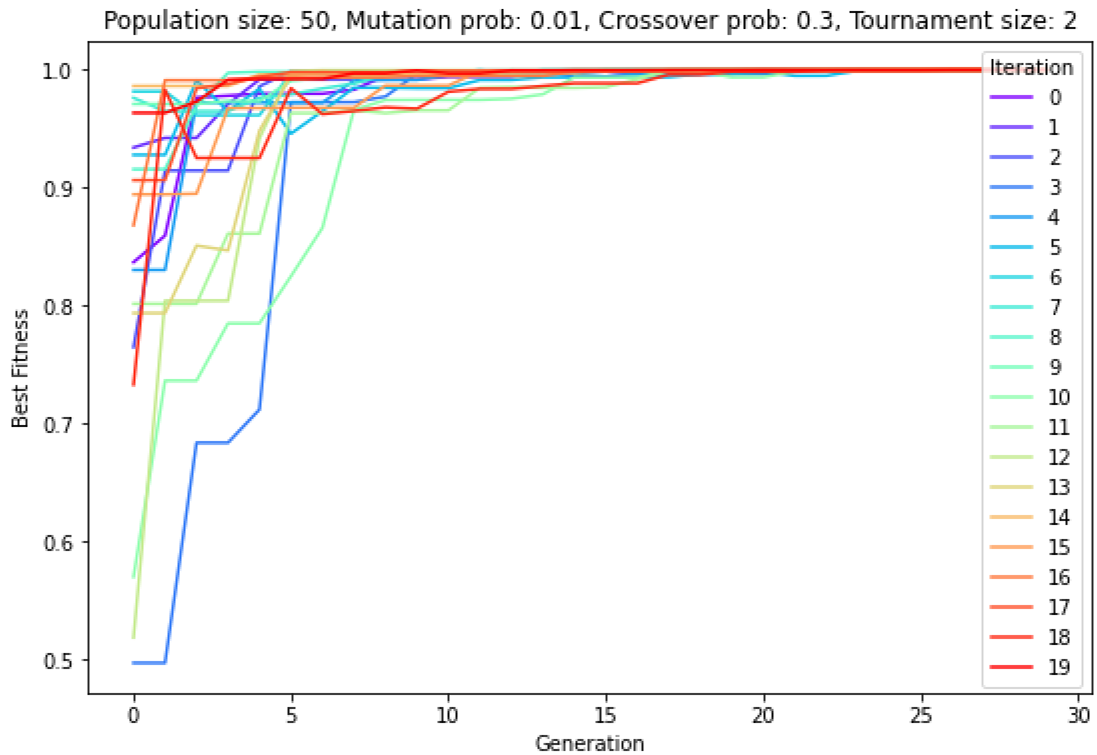

Figure 2. Best fitness

A solution (fitness = 1) is usually found within 5 generations.

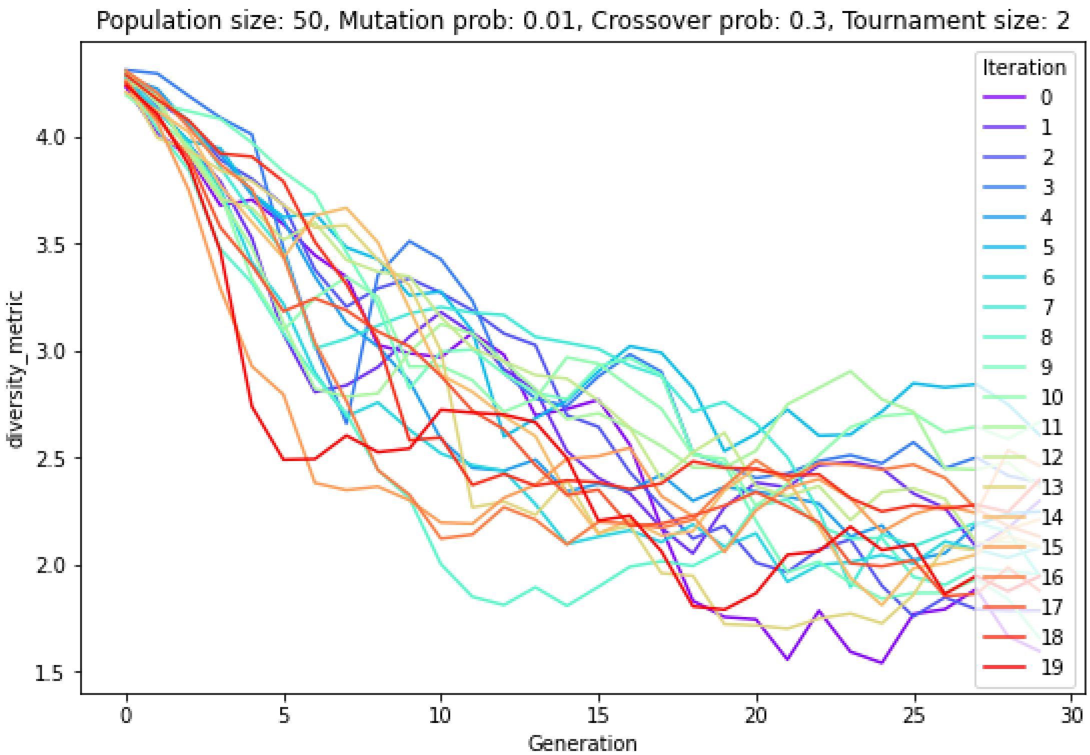

Figure 3. Diversity

There is a large spread in diversity, but it seems to converge in 20 generations and trends downward.

Vary Default - Average Fitness

The following figures show each of the 4 parameters being varied across the parameter space while the other parameters remained fixed at the default. The metric, average fitness, was averaged across the 20 iterations. Figure 1 and 2 both vary population size for average fitness, but figure 1 plots every iteration and figure 2 plots the average across iteration. Since the trends are similar, I decided to average to also smoothen out potential outliers.

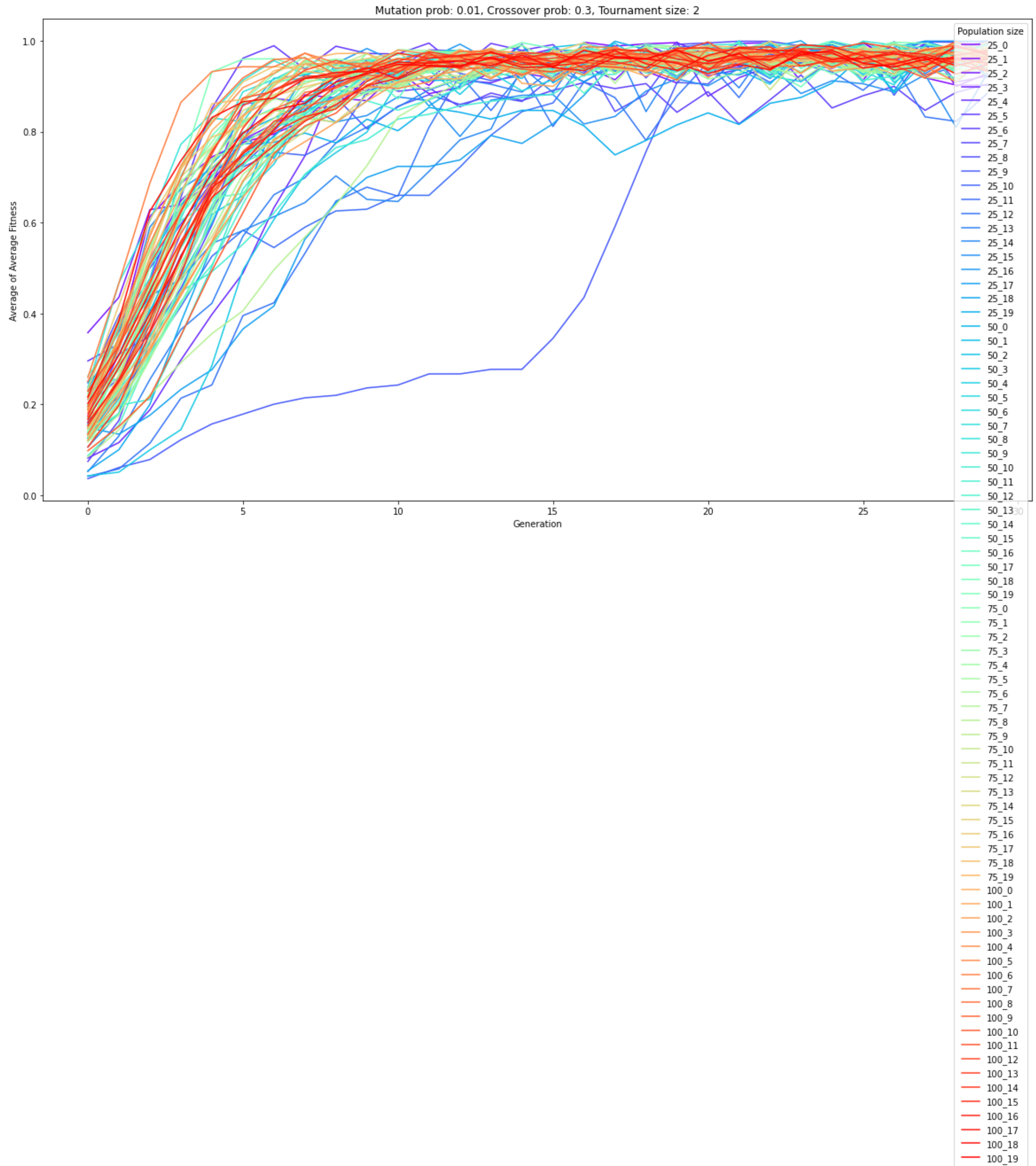

Figure 4. Varying population size with default parameters and all iterations plotted

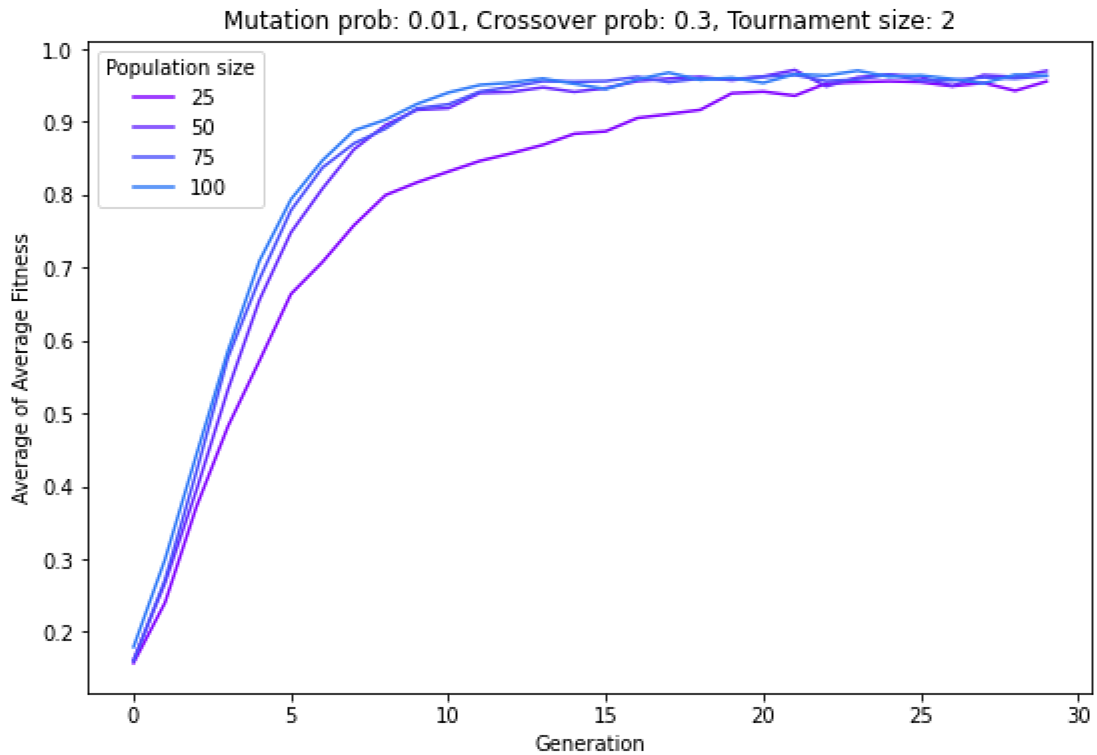

Figure 5. Varying population size

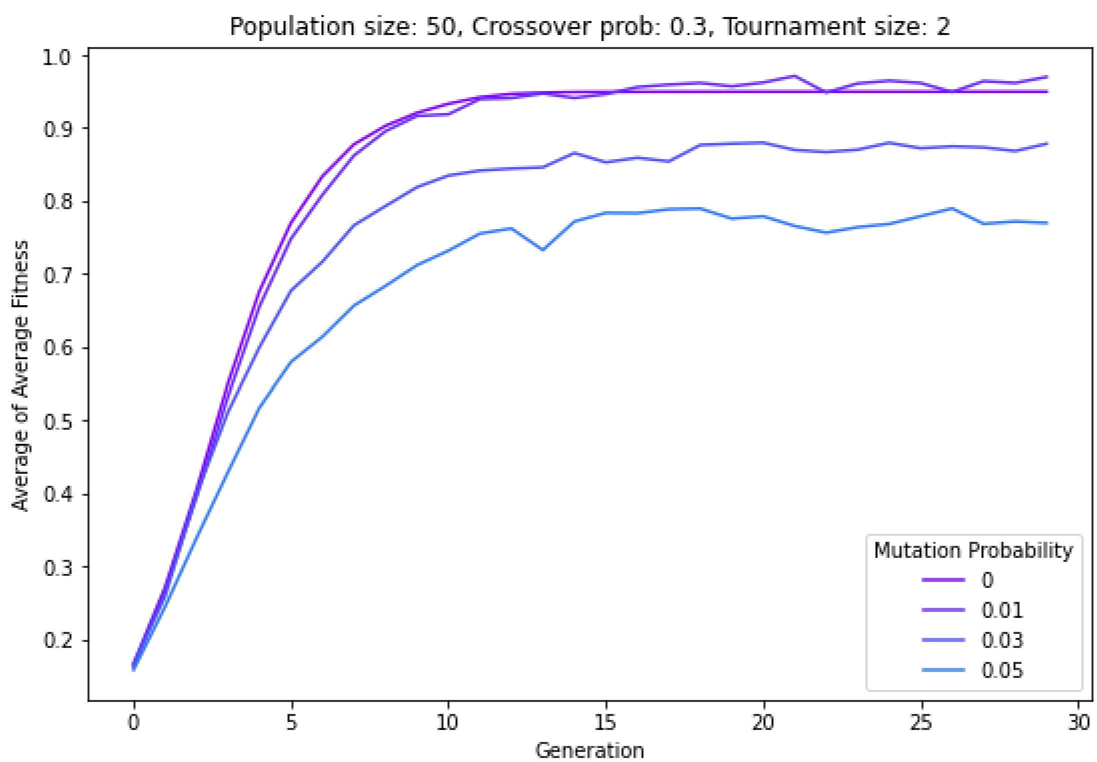

Figure 6. Varying mutation probability

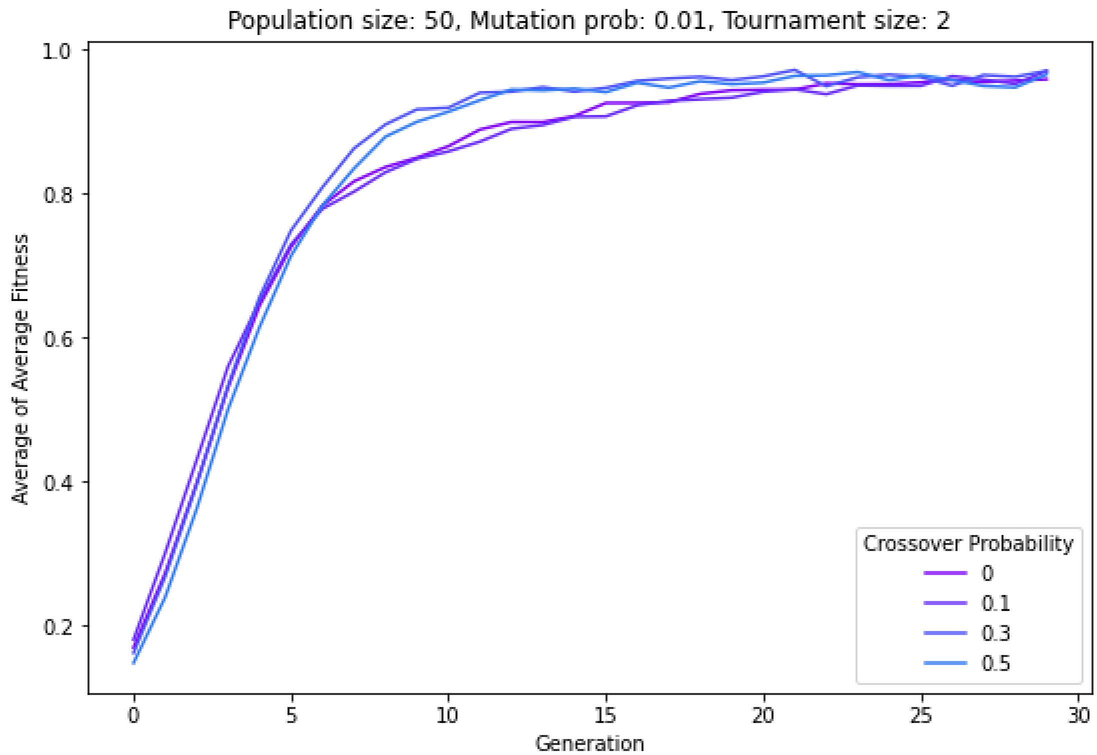

Figure 7. Varying crossover probability

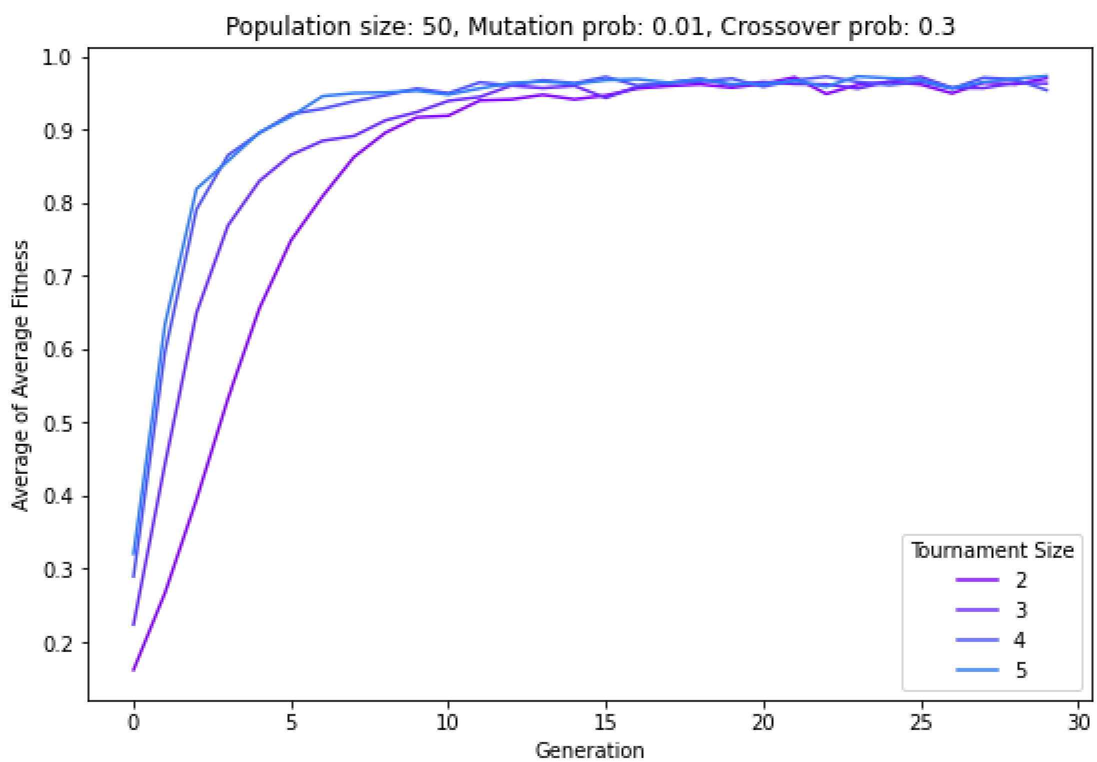

Figure 8. Varying tournament size

Vary default - Average of Best Fitness

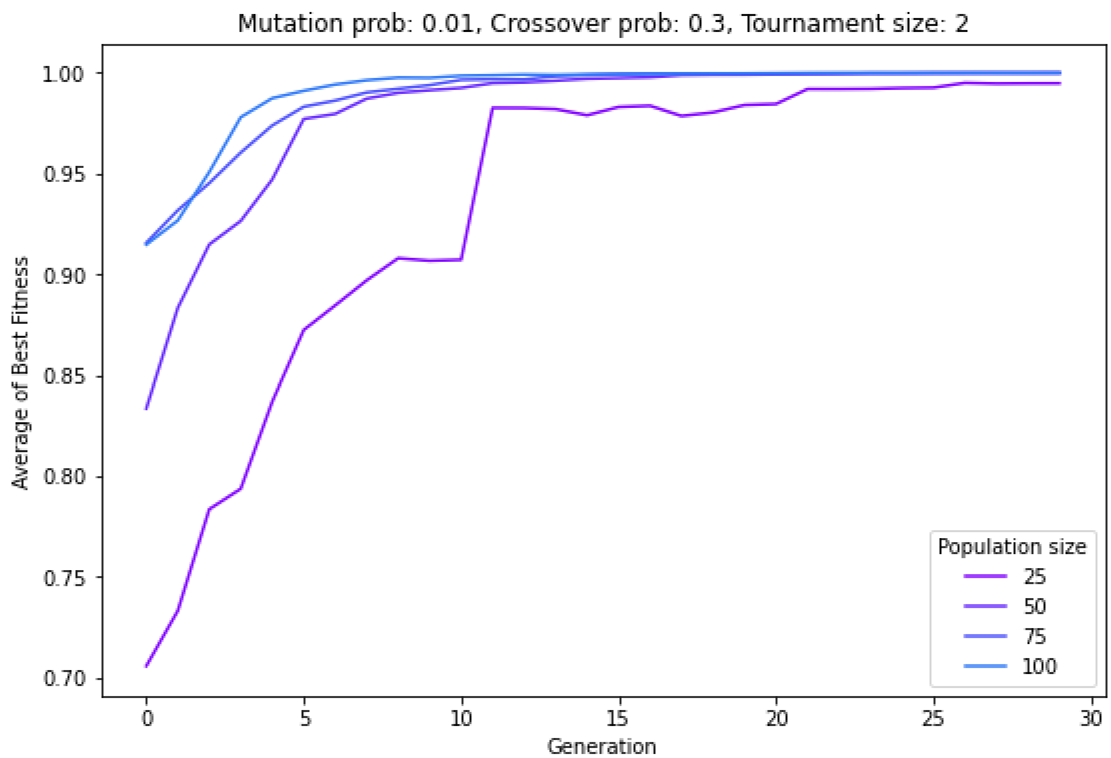

Figure 9. Varying population size

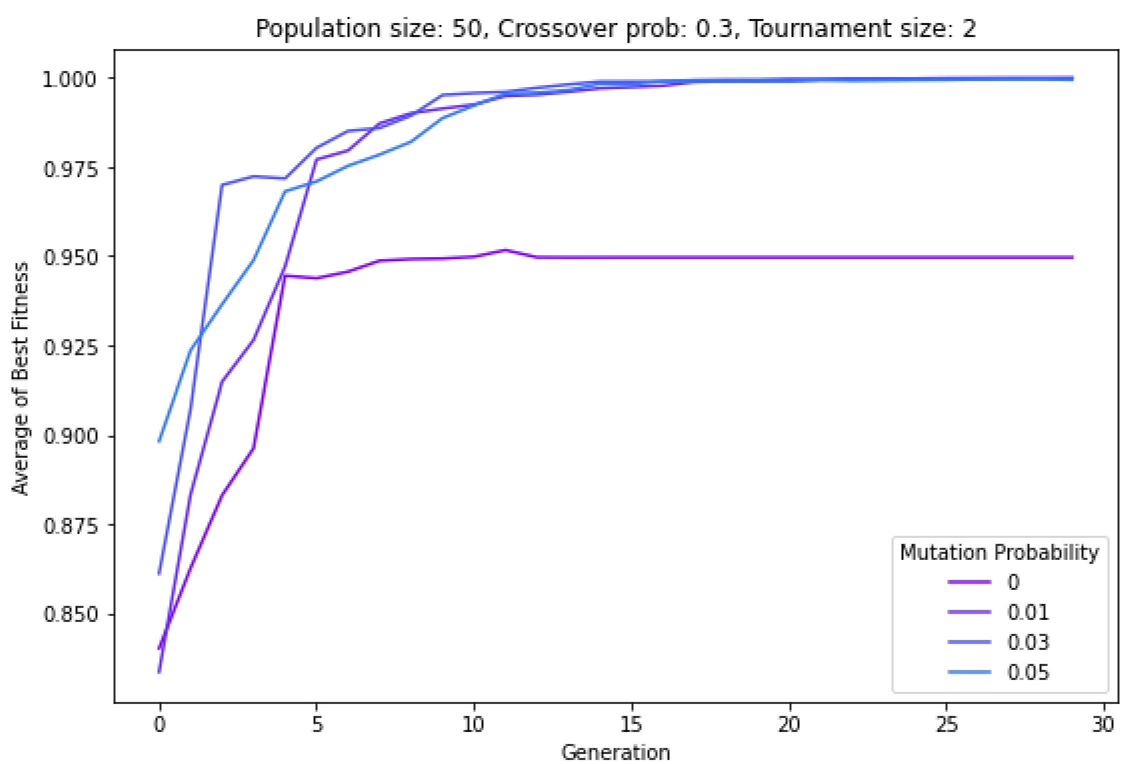

Figure 10. Varying mutation probability

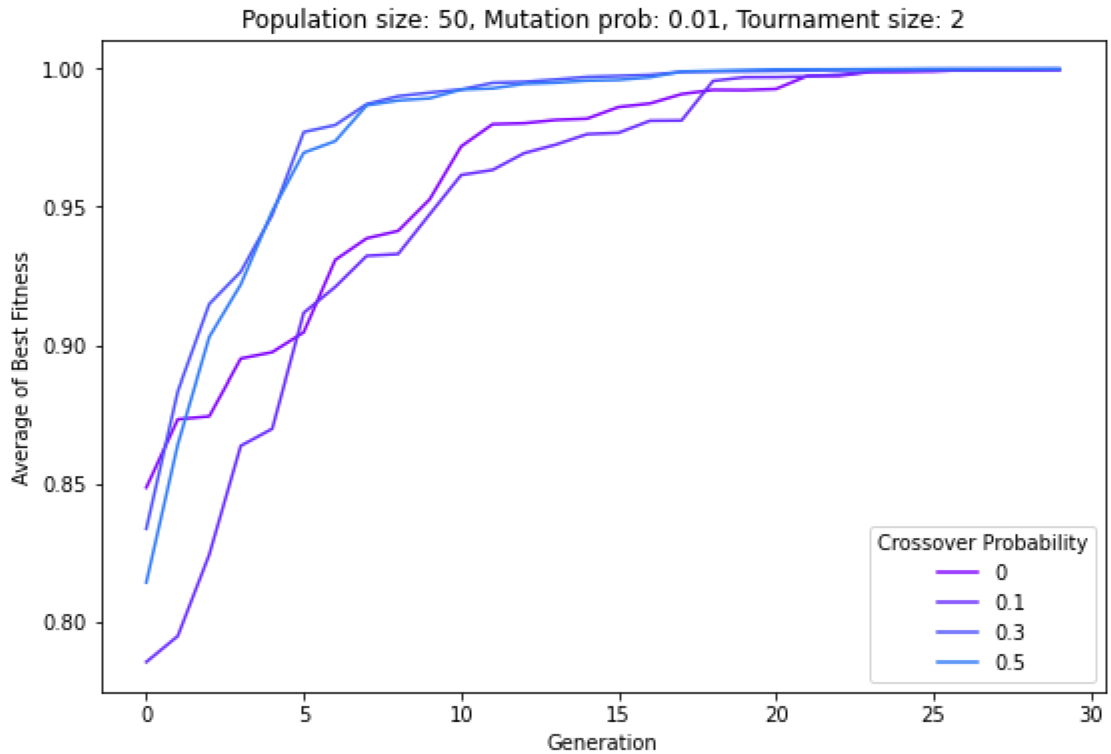

Figure 11. Varying crossover probability

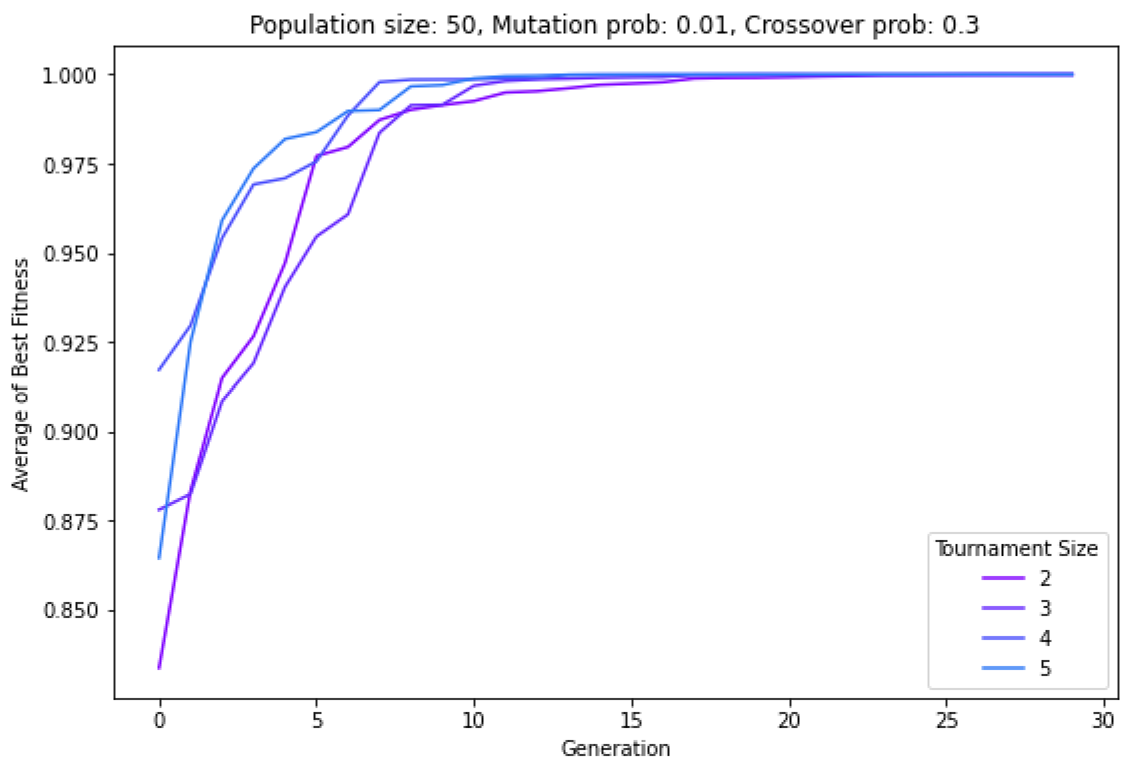

Figure 12. Varying tournament size

Vary default - Average of Diversity

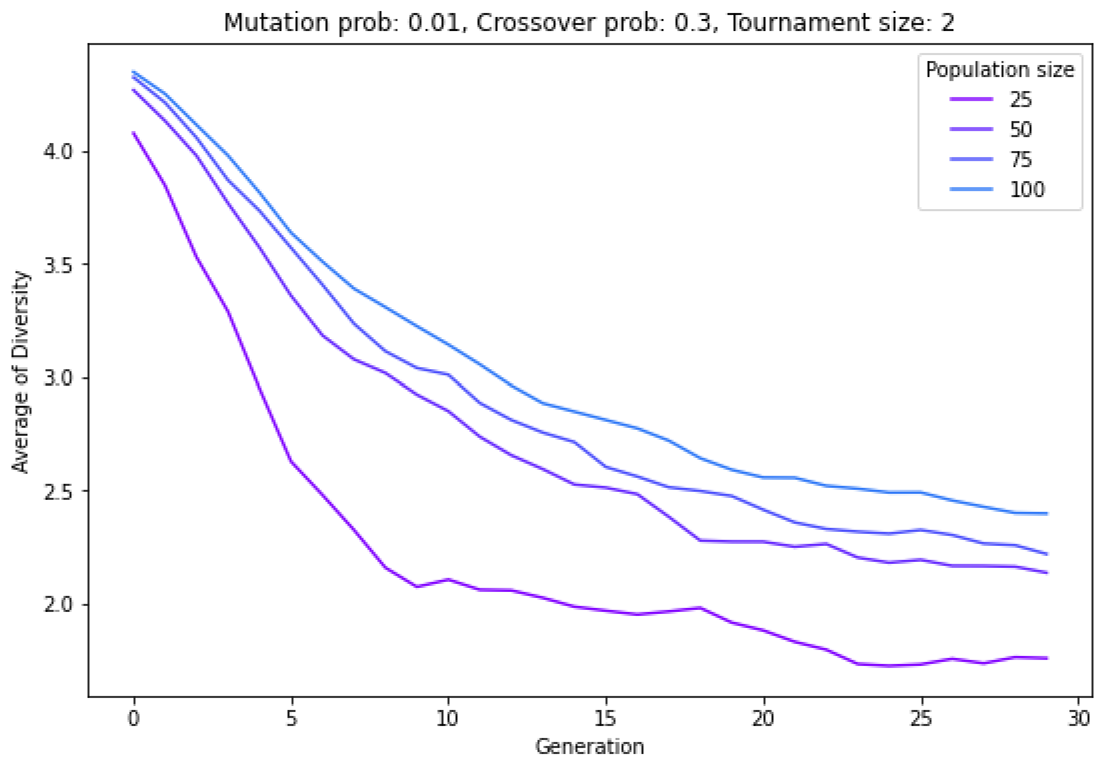

Figure 13. Varying population size

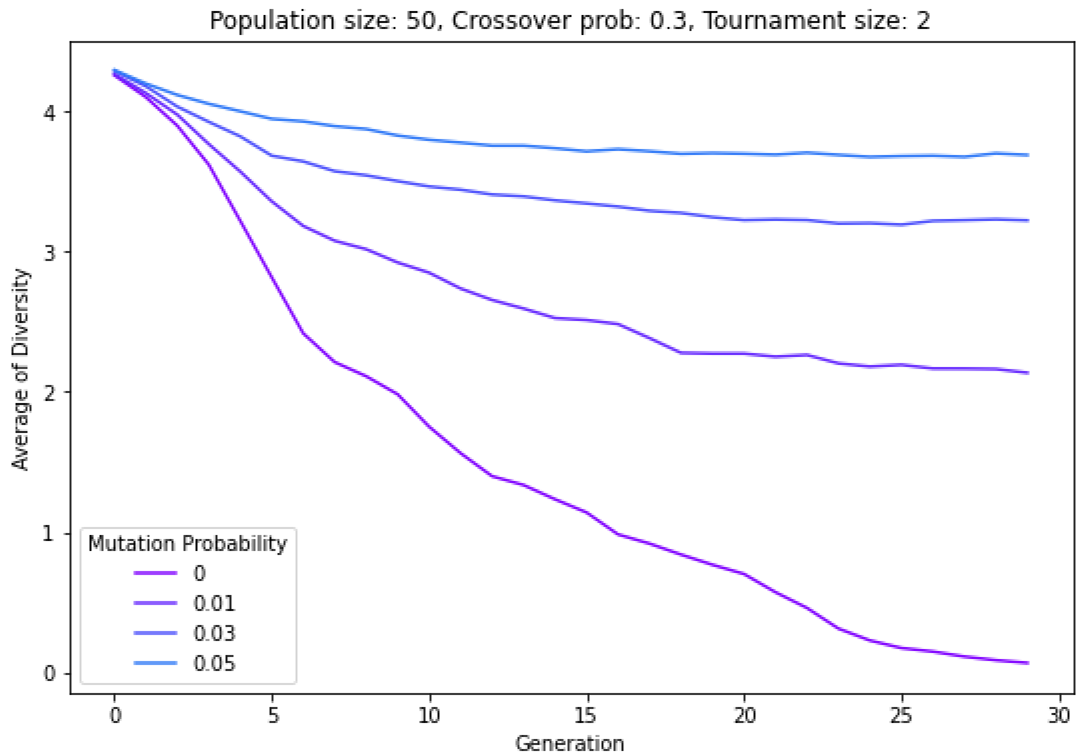

Figure 14. Varying mutation probability

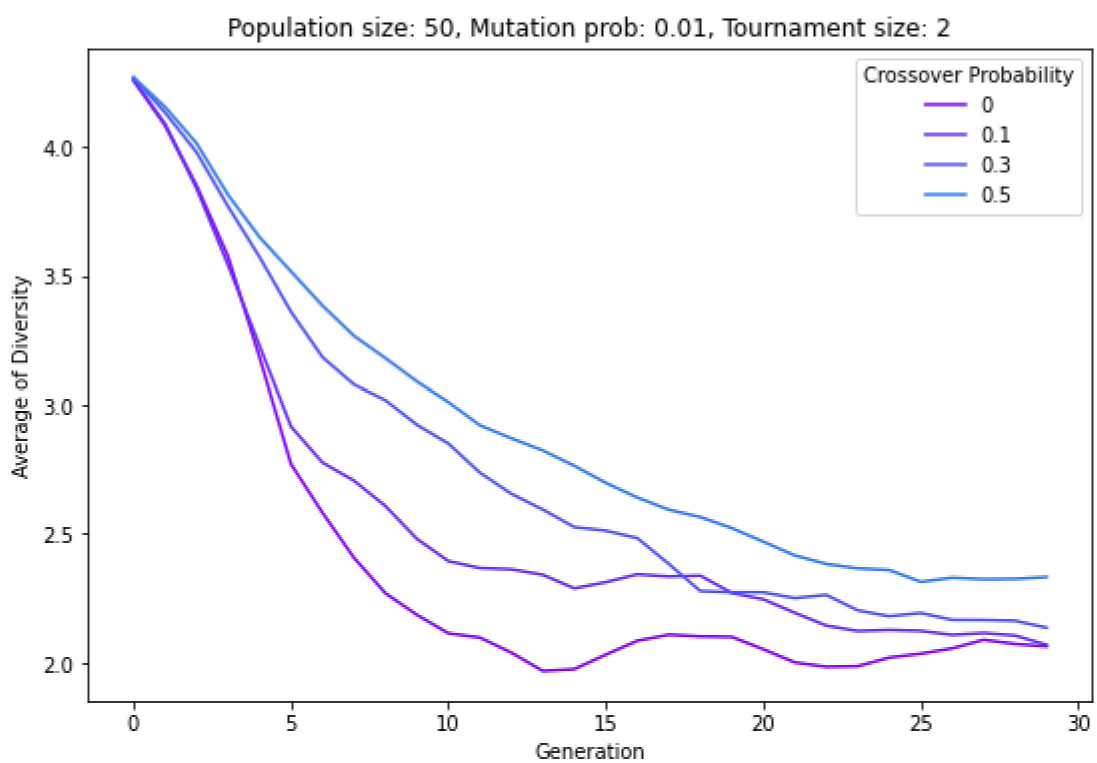

Figure 15. Varying crossover probability

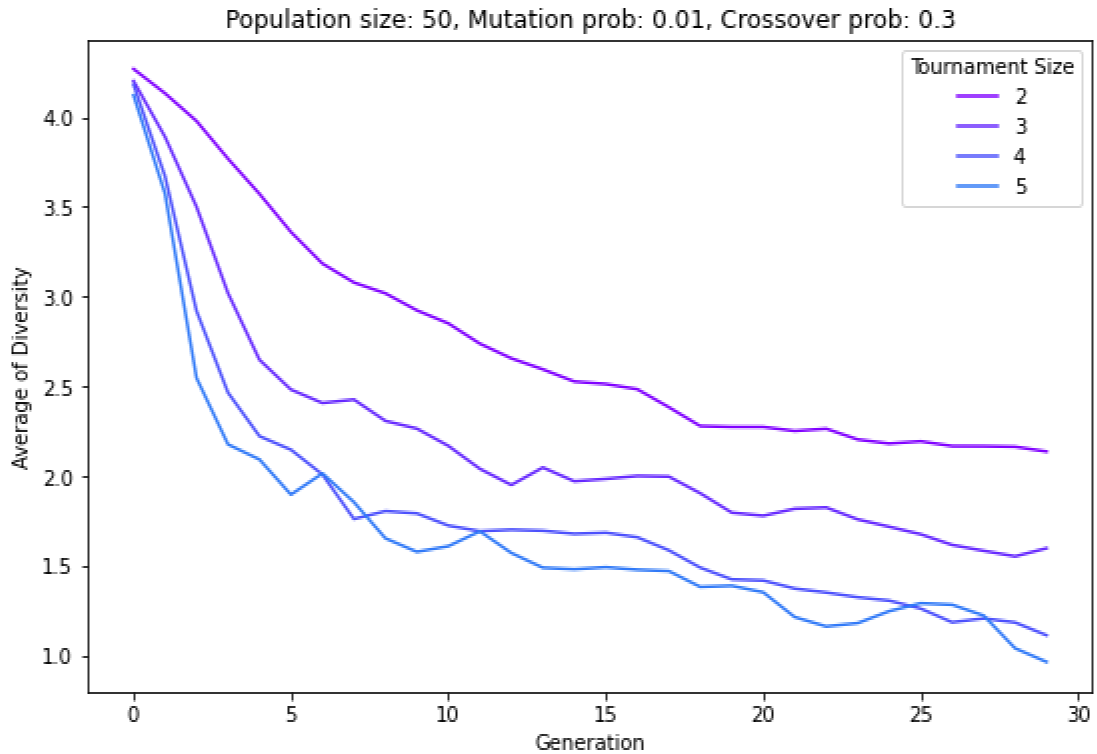

Figure 16. Varying tournament size

Vary 2 parameters, default - average fitness

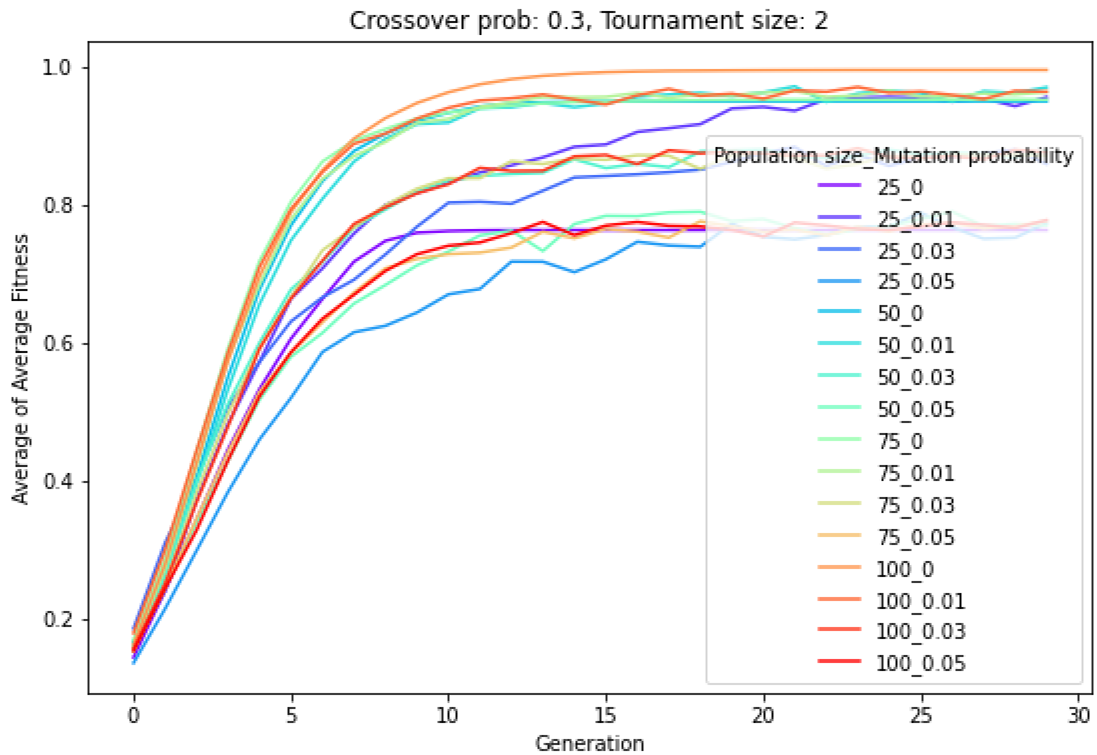

Figure 17. Vary population size and mutation probability

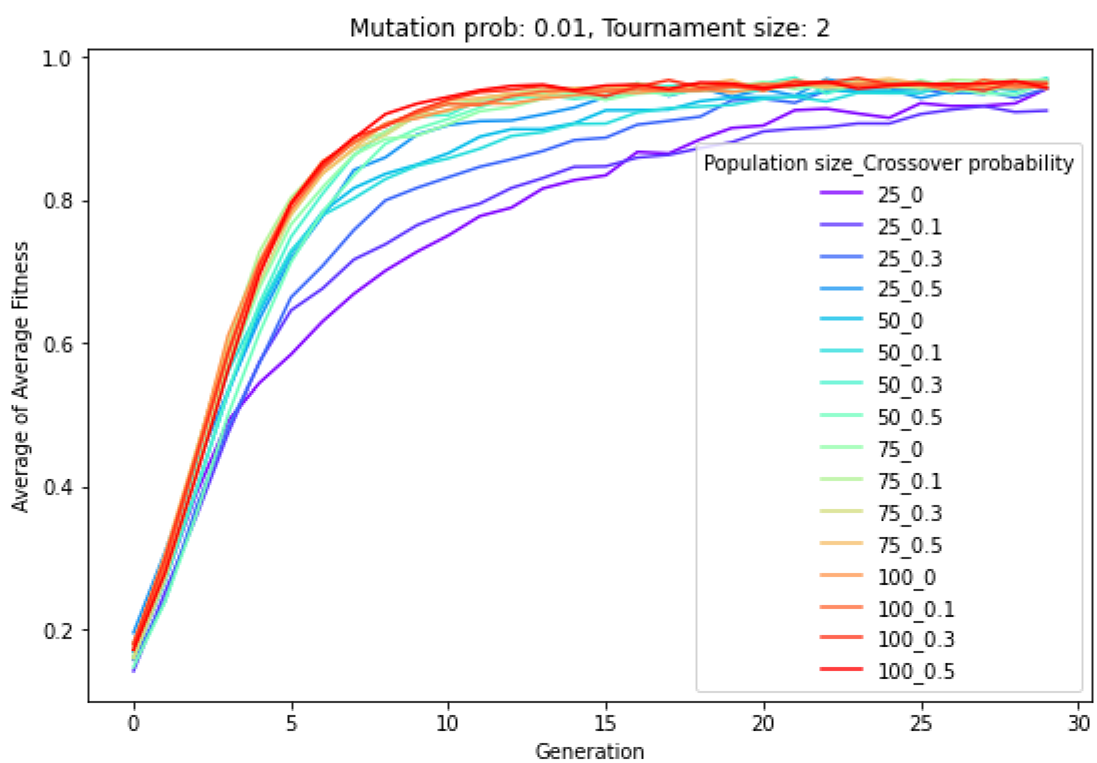

Figure 18. Population size and crossover probability

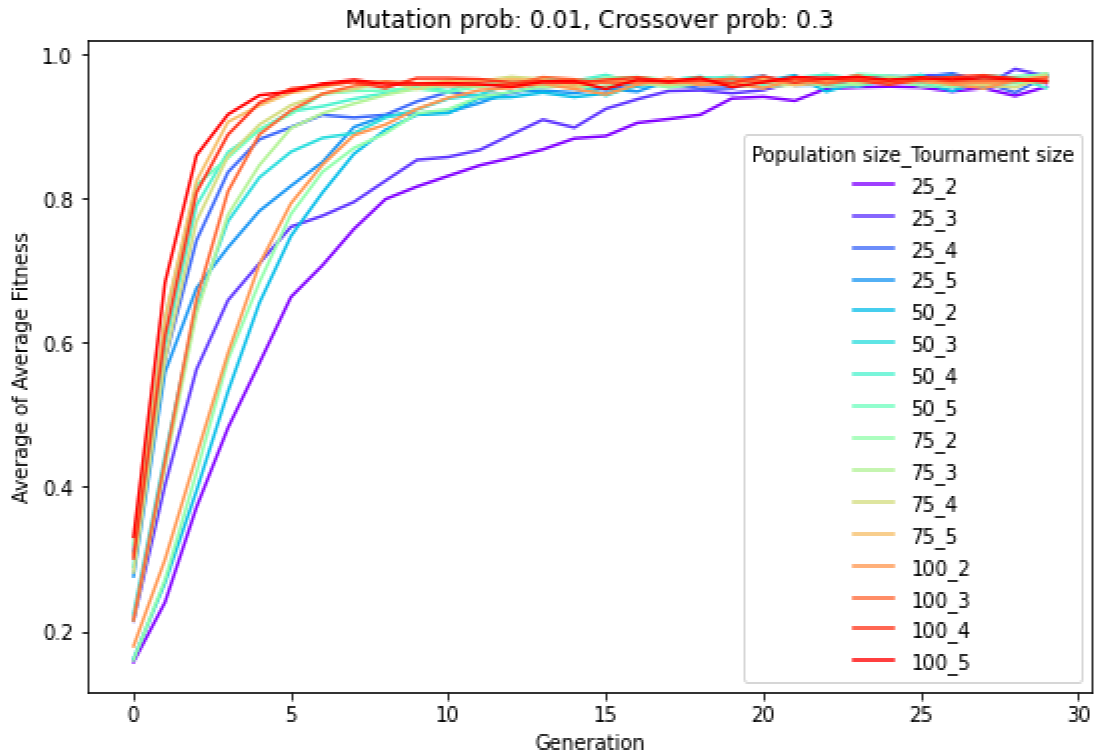

Figure 19. Population size and tournament size

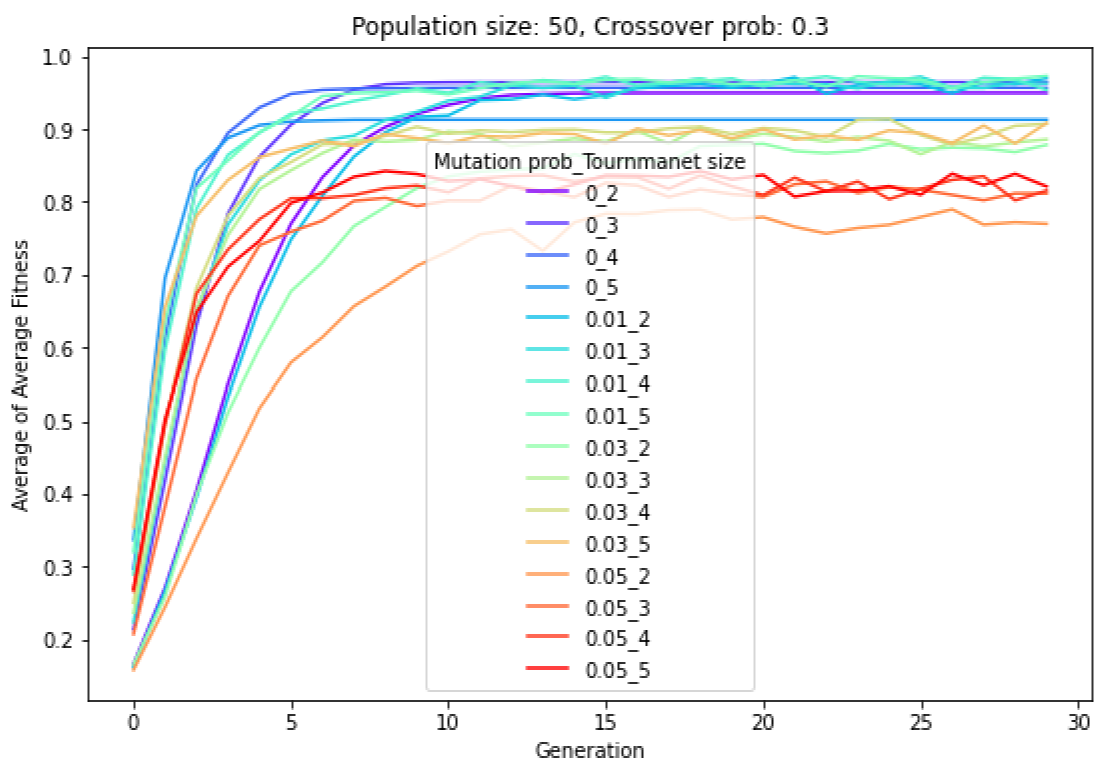

Figure 20. Mutation probability and tournament size

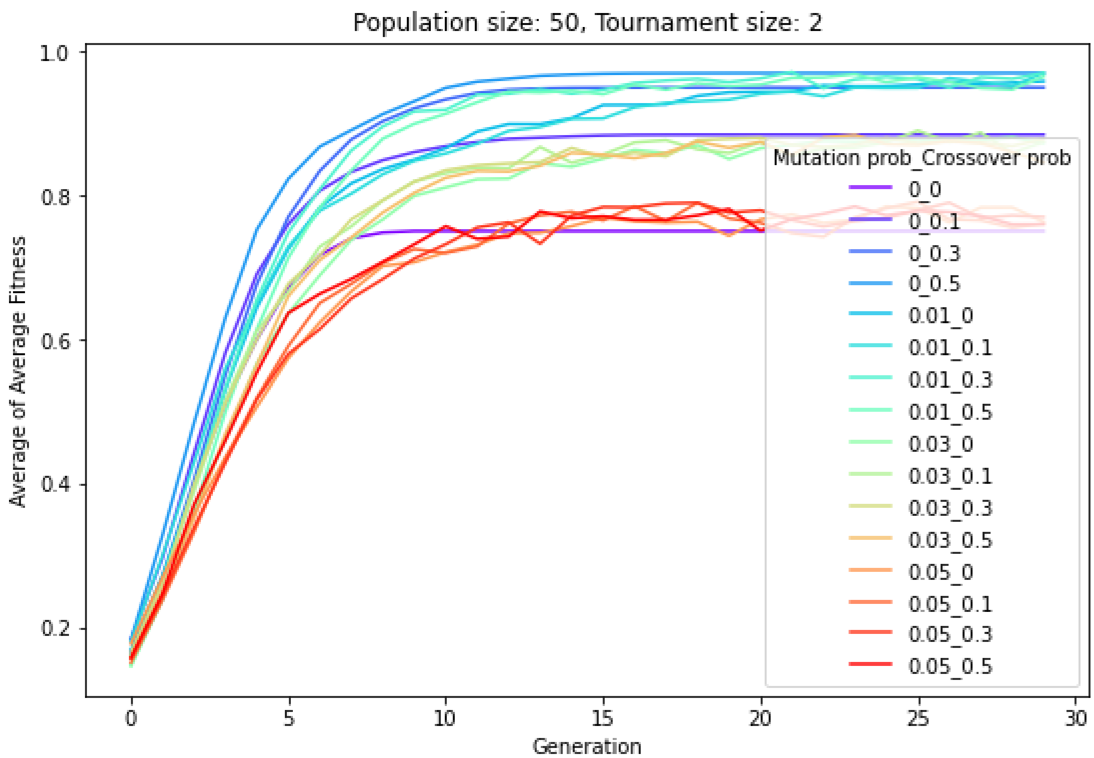

Figure 21. Mutation probability and crossover probability

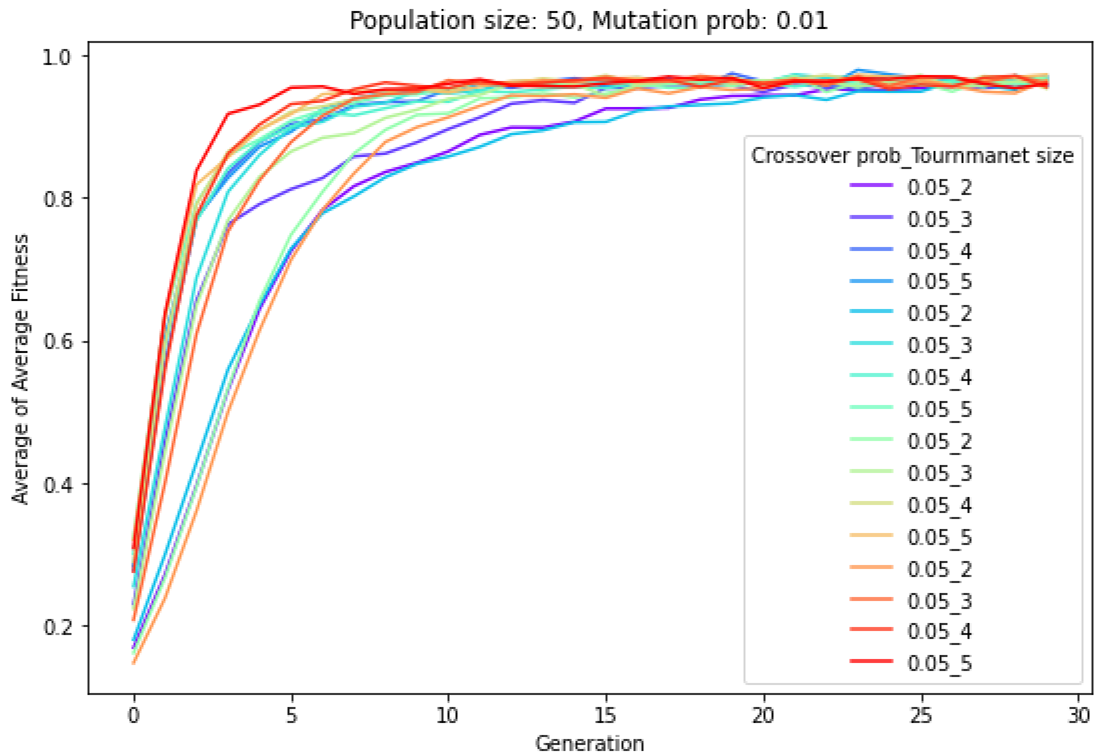

Figure 22. Crossover probability and tournament size

Discussion

The parameters that seemed to influence the fitness the most were mutation probability and, more marginally, population size. Figure 6 and figure 10 both show that for certain mutation probabilities, a fitness of 1 was never achieved when averaging over iterations. Figures 5 and 9 show that a smaller population size (mainly size of 25) made it hard for the algorithm to reach optimal fitness.

Higher selection pressure seems to allow the algorithm to converge faster. Figures 8 and 12 both show that, for higher selection pressure (higher tournament size), the algorithm reached higher values in fewer generations than with lower selection pressure. However, all levels of selection pressure did seem to converge to optimal fitness roughly the same time.

It appears that only either mutation or crossover is necessary. In figures 6 and 7, even with 0 mutation or 0 crossover, the algorithm reached optimal fitness. Figure 11 seems to support the same conclusion, but figure 10 shows that for 0 mutation, it only converged around 0.95, but that seems like an outlier. It makes sense intuitively since mutation and crossover are both means to allow for new information to enter the gene pool.

The parameters that seem to interact with each other the most are population size & mutation probability (Figure 17), mutation probability & tournament size (Figure 20), and mutation probability & crossover probability (Figure 21). All three figures show plots where some combination of these 2 parameters don't converge to the optimal fitness. They show much larger spread/variance in general suggesting that they influence each other strongly.

Figures 5 and 9 suggest that there is a certain population size that is required, but any greater will marginally allow for faster convergence. For example, a population size of 25 is too small and the algorithm has trouble converging, but a size of 100 is only marginally better than 75 or 50 for both average and best fitness.

In general, the higher the population size (figure 13), mutation probability (figure 14), and crossover probability (figure 15), the higher the diversity. However, a lower selection pressure (figure 16) improved diversity. This is in line with intuition since a larger population means more individuals to express different phenotypes, a higher mutation and crossover suggest more chances for new information to enter the gene pool, and a lower selection pressure means more chances for good offspring (but not the "best") to pass into the next generation.