Genetic Algorithm Optimization of the Booth Function

EARIN Lab 3 – Summer 2025
Students:
Cagla Kuleasan
Aayush Rautela

Variant 4: Optimize Booth Function using Roulette Wheel Selection

1. Task Description

The aim of this lab is to implement a genetic algorithm (GA) to optimize a two-dimensional mathematical function. For Variant 4, the function of interest is the **Booth function**, a standard benchmark for optimization algorithms, defined as:

$$f(x, y) = (x + 2y - 7)^2 + (2x + y - 5)^2$$

The GA must use **Roulette Wheel Selection** to choose parents and work over a domain of $x, y \in [-5, 5]$. The algorithm should be capable of converging to the global minimum, known to be at $(1, 3)$ with $f(1, 3) = 0$.

2. Genetic Algorithm Implementation

The following components were implemented:

- **Initialization**: Population initialized with random float values for x and y within the given bounds.
- **Fitness Function**: The fitness of an individual is calculated as $1/(1 + f(x, y))$, converting the minimization problem to a maximization problem.
- **Selection**: Roulette Wheel Selection based on normalized fitness scores.
- **Crossover**: Random interpolation ($\alpha \cdot p_1 + (1 - \alpha) \cdot p_2$) with a configurable rate.
- **Mutation**: Gaussian noise added to x and y values, governed by mutation rate and strength.

The algorithm was evaluated over 100 generations.

3. Experiments and Results

Experiment 1: Parameter Tuning

Several combinations of GA parameters were tested to identify configurations yielding the best fitness values.

Combination	Population Size	Mutation Rate	Mutation Strength	Crossover Rate	Best Fitness	Best Solution (x, y)
1	30	0.1	0.05	0.7	0.337916	(1.9266, 1.9710)
2	50	0.2	0.10	0.8	0.369892	(1.9461, 2.1072)
3	70	0.3	0.10	0.9	0.397582	(1.8838, 2.1451)
4	**100**	**0.1**	**0.20**	**0.6**	**0.997673**	**(1.0015, 2.9772)**
5	50	0.4	0.05	0.8	0.355697	(1.9841, 2.0959)

Conclusion: Combination 4 performed the best and was selected for use in the following experiments.

Experiment 2: Random Seeds and Population Size

To assess the impact of randomness, the best configuration was run using 5 different seeds:

- **Best Fitness Across Seeds**: 0.997673
- **Average Fitness**: 0.986662
- **Standard Deviation**: 0.016859

Then, the population size was decreased to observe its effect:

Population Size	Best Fitness
50	0.940074
25	0.460147
10	0.330494

Conclusion: Lower population sizes reduce solution quality significantly.

Experiment 3: Crossover Rate Impact

The crossover rate was varied and results averaged over three seeds:

- **Best performance** at crossover rate **0.5**
- Higher rates (0.8, 1.0) led to more volatile convergence and weaker average fitness

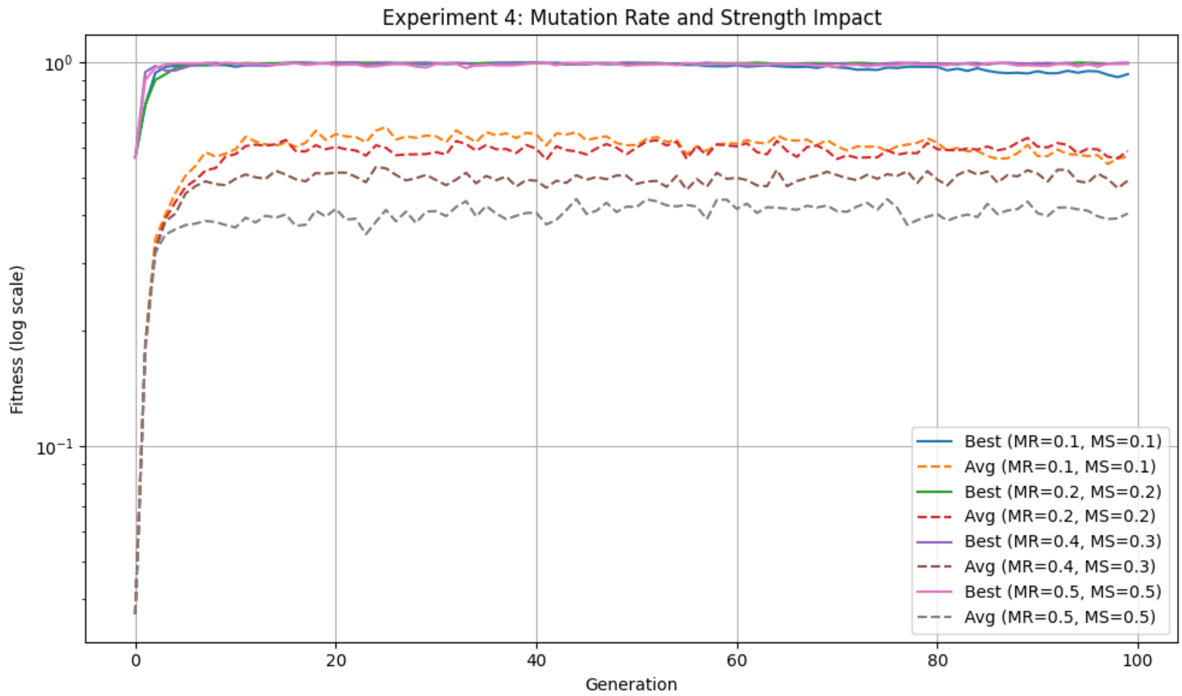
A log-scaled plot was generated to visualize these trends.

Experiment 4: Mutation Rate and Strength Impact

Four combinations of mutation rate and strength were tested:

- Best results were observed with **mutation rate = 0.2** and **strength = 0.2**
- Higher mutation values (e.g., 0.5) led to decreased average performance

This confirmed that over-mutation harms convergence despite encouraging exploration.



4. Insights and Reflections

This lab highlighted the delicate balance between exploration and exploitation in genetic algorithms. Some key takeaways:

- Moderate values of mutation and crossover yield the best results
- Roulette Wheel Selection effectively maintains diversity while promoting fit individuals
- Running experiments across seeds is essential for understanding stability
- Visualizing convergence behavior across generations is crucial for parameter analysis

Through this lab, the following learning goals were met:

- Understanding the flow and components of a genetic algorithm

- Recognizing the importance of fitness evaluation and proper selection mechanisms
- Gaining intuition on how mutation and crossover affect evolutionary behavior
- Learning to interpret fitness graphs, variance across runs, and parameter sensitivity
- Developing critical thinking by running real experiments and comparing outcomes
- Applying theoretical algorithm knowledge to a practical optimization task

Overall, this lab deepened my understanding of evolutionary algorithms and how parameter tuning significantly influences performance.