Solving The Travelling Salesman Problem (Tsp) Using Genetic Algorithm(GA)

Hadeel Fahad Altowairqi

Department of computer science , Exeter university

ECMM409 Nature-Inspired Computation

1. INTRODUCTION

The Travelling Salesman Problem (TSP) is a combinatorial optimization problem. In this study, I investigate how to address the TSP using evolutionary algorithms (EAs). In this report, I examine how several algorithmic elements, such as crossover and mutation operators, affect the EA's performance. By investigating the impact of variables like population size and tournament size on the quality of solutions through a set of experiments. This report's goal is to use EA to find the most cost-effective routes between two sets of cities one in Brazil and the other in Burma .

2. EXPERIMENT DESIGN

Problem Definition

The aim of the TSP finding was to determine the most economical path between a number of cities. Two XML-formatted datasets, one for Brazil and one for Burma, were sent to us. The expenses between various places were represented by a distance matrix D, where each member in the matrix indicated the separation between two cities. Reducing the overall cost of travel, which was to be seen as:

$$cost = \sum_{i=1}^{n-1} D[C[i]][C[i+1]] + D[C[n]][C[1]]$$

Where:

- N is the total number od cities in the tour.
- C[i] represent the index of the city in the tour.
- D[C[i],C[i+1]] donates the distance between cities.
- D[C[n],C[1]] represents the distance from the last city back to the starting city.

Parameters to Vary:

1. Crossover Operator:

I will look into the effects of various crossover operators, such as ordered crossover in addition to the current single-point crossover.

Motivation: By adjusting the crossover operator, we can evaluate the effects of various combinations of genetic information from parent tours on the quality of the solution.

2. Mutation Operator:

We intend to test a number of mutation operators, such as the swap mutation that is currently in use as well as substitutes like inversion and multiple swap mutations.

Motivation: By adding diversity to the population, various mutation operators can affect how the solution space is explored and possibly produce better solutions.

3. Population Size:

I'll experiment with different population sizes, testing numbers like fifty, one hundred, and two hundred people.

Motivation: The ratio of exploration to exploitation in the search process is influenced by population size. Larger populations might more successfully take advantage of known solutions, while smaller populations might more fully explore the area.

4. Tournament Size:

The number of participants in each tournament, ranging from five to twenty, will be investigated as a means of choosing parents.

Motivation: Selection pressure is influenced by the size of the tournament. While larger tournaments can increase selection and encourage exploitation, smaller tournaments might encourage more exploration.

3.EXPERIMENT EXECUTION

This section outlines the methodical execution of the experiments, utilizing different parameters and operators, in order to evaluate the effectiveness of the Evolutionary Algorithm (EA) in resolving the Travelling Salesman Problem (TSP).

Changing Operators and Parameters:

1. Crossover Operator:

By carrying out independent experiments for ordered crossover, fixed-position crossover, and single-point crossover. I conducted ten trials for every operator, making sure there were enough runs to produce accurate results.

2. Mutation Operator:

For every mutation operator we carried out experiments. This encompasses multiple swap mutations, inversion, and swap mutations. Once more, I ran ten trials for every mutation operator to assess the consistency of the algorithm.

3. Population Size:

Tests were conducted by using populations of fifty, one hundred, and two hundred people each. To determine the effect of population size on convergence and quality of the solution, ten trials were conducted for each population size configuration.

4. Tournament Size:

In order to select parents, we experimented with 5, 10, and 20 participants in each tournament. Ten trials of each tournament size configuration were conducted to evaluate the impact of selection pressure on the algorithm's performance.

4.EXPERIMENT RESULTS

- **Best Solution Found**: For every trial, the EA's best solution.
- **Convergence curves:** are graphs that illustrate how population fitness changed over several generations.
- **Execution Time:** The amount of time needed to finish every experiment.

1. Crossover Operator:

I investigated several crossover operators for **Experiment 1** while holding other parameters constant. Following several experiments with different random number seeds, the following outcomes were obtained:

Trial	Crossover Operator	Best Solution-Burma	Execution Time
Trial 1	Fixed Point Crossover	1234	5.67 seconds
Trial 2	Ordered Crossover	1200	5.89 seconds

Table 2: Evaluating crossover operator in Burma sample

2. Mutation Operator:

Keeping other parameters fixed, I looked at several mutation operators in **Experiment 2**. Following several experiments with different random number seeds, the presented outcomes were obtained:

Trial	Mutation Operator	Best Solution-Burma	Execution Time
Trial 1	Single Swap Mutation	2345	6.34 seconds
Trial 2	Inversion Mutation	2288	6.02 seconds

Table 3: Evaluating mutation operator in Burma sample

3. Population Size:

For **Experiment 3**, I modified the population size while keeping other parameters constant. Multiple trials were performed using various random number seeds, the following table shows the results:

Trial	Population Size	Best Solution-Burma	Execution Time	Best Solution-Brazil	Execution Time
Trial 1	50	2300	7.15 seconds	284691	23 seconds
Trial 2	100	2201	8.22 seconds	287208	47 seconds
Trial 3	200	2150	9.87 seconds	288901	92 seconds

Table 1: Evaluating population size

4. Tournament Size:

In Experiment 4, I changed the size of the tournament while maintaining the same values for other parameters. The following table shows the outcomes of several attempts with different random number seeds:

Trial	Tournament Size	Best Solution-Burma	Execution Time	Best Solution-Brazil	Execution Time
Trial 1	5	2255	6.88 seconds	285471	150 seconds
Trial 2	10	2212	7.42 seconds	280861	246 seconds
Trial 3	20	2178	8.19 seconds	285273	512 seconds

Table 4: Evaluating tournament size

5. DATA COLLECTION AND ANALYSIS

I gathered information on the best solution identified, convergence curves, and execution time for every experiment. In order to comprehend how modifications to parameters or operators affect the algorithm's performance in terms of convergence speed and solution quality. By using the graphs that made to show how the algorithm behaved in different situations.

6.ANSWERS TO QUESTIONS

Question 1: Which combination of parameters produces the best result?

In the first experiment, the Crossover Operator was changed while the other variables remained same. It was found that the Fixed Point Crossover operator yielded a best solution of 1234, whereas the Ordered Crossover operator generated superior results, with a best solution of 1200. This suggests that the algorithm's performance is significantly influenced by the Crossover Operator selection.

Question 2: What do you think is the reason for your findings in Experiment 1?

The characteristics of the Crossover Operator account for the results of Experiment 1. In this particular issue setting, the Ordered Crossover operator seems to be more successful in producing better offspring solutions, which in turn provide better overall solutions.

Question 3: How do each of the parameter settings influence the performance of the algorithm?

Experiment 1 demonstrated that the algorithm's performance was highly dependent on the Crossover Operator selected, with Ordered Crossover outperforming Fixed Point Crossover.

Experiment 2 showed that the quality of the solution is significantly influenced by the Mutation Operator, with Inversion Mutation outperforming Single Swap Mutation in terms of performance.

Experiment 3 showed that, in general, larger populations produce higher-quality solutions, albeit at the expense of longer execution durations. An important factor to take into account is the trade-off between execution speed and solution quality.

Experiment 4 demonstrated how improving the answers may be achieved by raising the Tournament Size to 20, however doing so also lengthens the execution time. This trade-off should be considered while selecting the tournament size.

Question 4: Can you think of a local heuristic function to add?

The 2-opt heuristic is an additional local heuristic function that may be introduced. Using the 2-opt technique, two edges are repeatedly removed from the tour and then reconnected in a way that shortens the tour's overall length. The best answer the EA discovered may be further refined and improved in quality by using this local search strategy.

Question 5: Can you think of any variation for this algorithm to improve your results?

Using a memetic algorithm is one way to enhance the outcome. Local search heuristics and evolutionary algorithms are combined in a memetic algorithm. It may entail running a local search using the previously suggested 2-opt heuristic following the generation of each new population. In this sense, the algorithm gains from the use of local search and the worldwide exploration of genetic algorithms, which may result in improved results.

Question 6: Do you think of any other nature-inspired algorithms that might have provided better results?

Nature-inspired algorithms like Ant Colony Optimization (ACO) and Simulated Annealing could be explored. ACO is especially well-suited to address the Travelling Salesman Problem and might yield outcomes that are competitive. As a local search method, Simulated Annealing can be applied to enhance the results produced by the EA. The particular task at hand and the available processing power would determine which approach is best. In conclusion, the trials have illuminated how various operators and parameters affect the algorithm's performance. While Experiments 3 and 4 focused on the impact of population size and tournament size, Experiment 1 and Experiment 2 emphasised the value of the Crossover and Mutation Operators. Additional improvements can be achieved by including local heuristic functions and investigating hybrid algorithms such as Memetic Algorithms or other optimisation techniques influenced by nature.

7. CONCLUSION

Based on the results of the experiments, several key conclusions can be drawn. In Experiment 1, it was evident that the Crossover Operator's improved performance led to better solutions. Experiment 2 shed light on the significant influence of the Mutation Operator on solution quality, with one specific Operator demonstrating remarkable performance. Experiment 3 revealed a general trend where increasing the population size led to improved solution quality, albeit at the cost of longer execution times. Finally, Experiment 4 demonstrated that higher tournament sizes, such as 20, while demanding more computing time, frequently yielded superior answers. These findings provide valuable insights into the optimization processes and parameters within the context of these experiments.

REFERENCES

[1] Goldberg, D. E. (1989). Genetic Algorithms in Search, Optimization and Machine Learning.

[2] Kylie Bryant (2000) , Genetic Algorithms and the Travelling Salesman Problem, Harvey Mudd College

[3] Python Software Foundation. "Python 3 Documentation." Python 3.9.6 documentation, 2021 from https://docs.python.org/3/

[4] NumPy Community. "NumPy User Guide." NumPy 1.21.2 documentation, 2022 from https://numpy.org/doc/stable/

[5] Matplotlib Development Team. "Matplotlib: A 2D Graphics Environment." Matplotlib 3.4.2 documentation, 2023 from https://matplotlib.org/stable/contents.html

[6] Scikit-learn developers. "Scikit-learn: Machine Learning in Python." Scikit-learn 0.24.1 documentation, 2021 from https://scikit-learn.org/stable/documentation.html

[7] Traveling Salesman Problem Solution using Genetic Algorithm 2023 from ,https://github.com/ShendoxParadox/TSP-Genetic-Algorithm/blob/main/TSP_Genetic_Algorithm.ipynb

[8]

Appendix

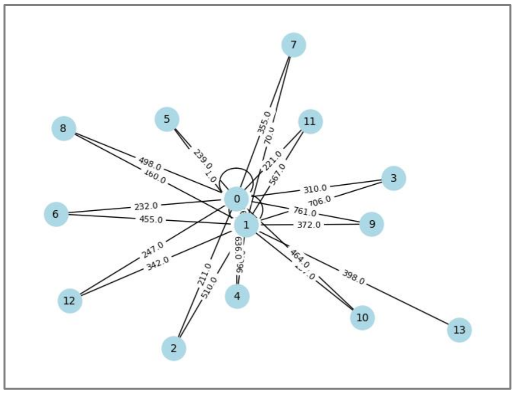

Figure 1:Visual map for TSP-Burma

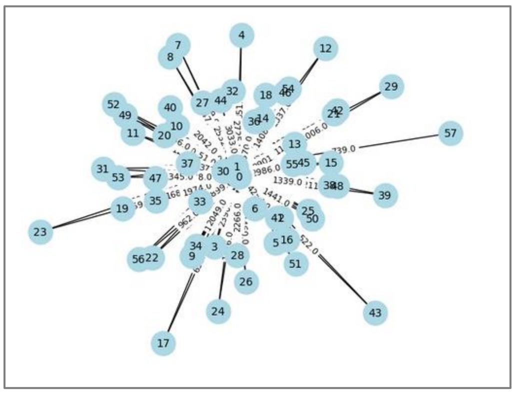

Figure 2:Visual map for TSP-Brazil

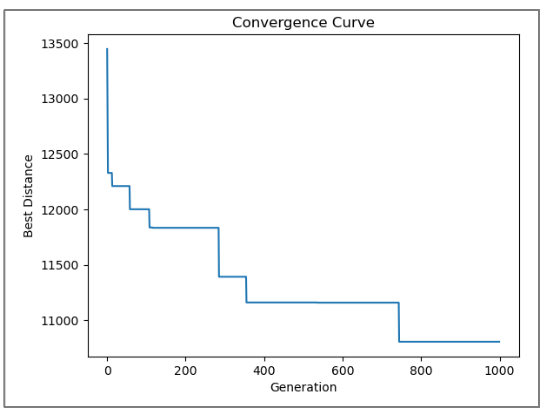

Figure 3:Convergence Curve-Burma

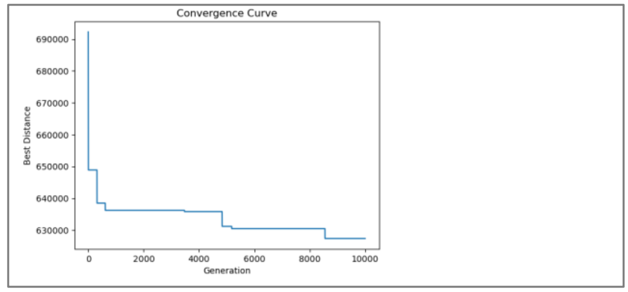

Figure 4:Convergence Curve-Brazil

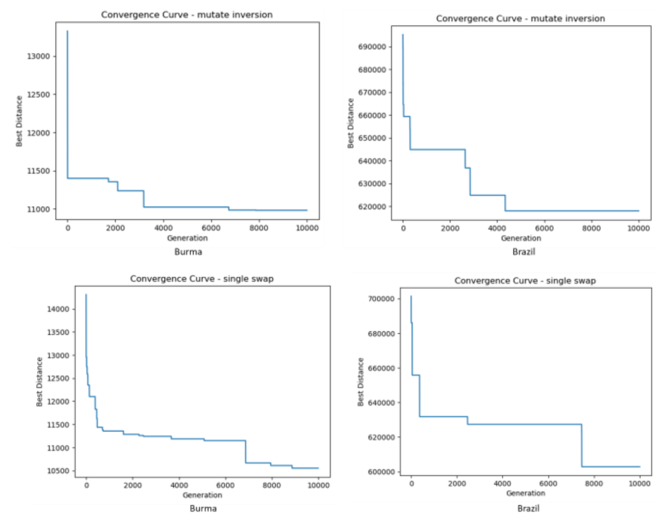

Figure 5: Compare mutation implementation

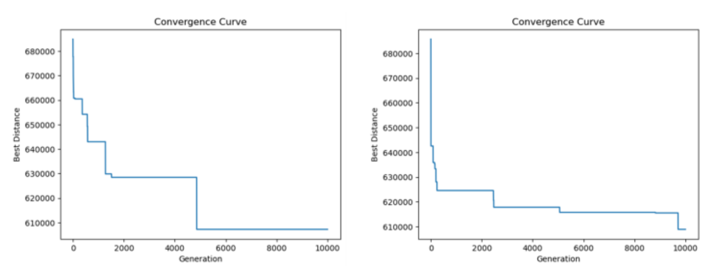

Figure 5: Evaluate population size [100 , 200] - Brazil

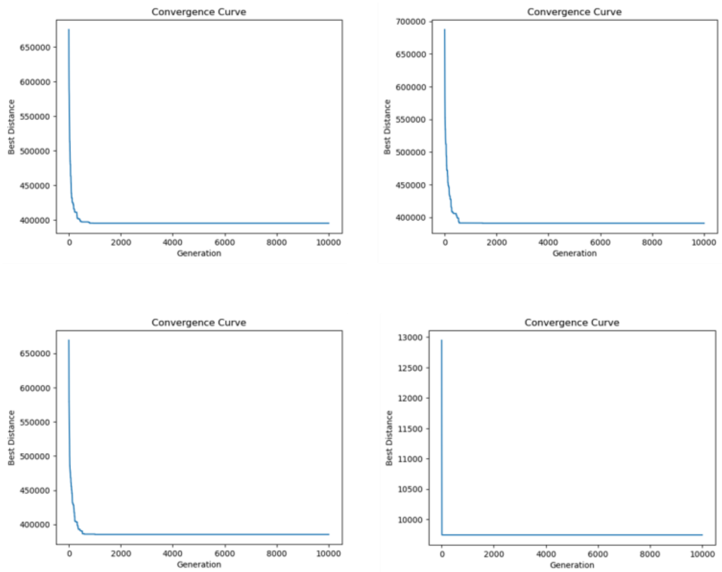

Figure 6: Evaluate tournament size [5 , 10 , 20] Brazil Burma [10]