JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)
Journal homepage: https://jurnal.stkippgritulungagung.ac.id/index.php/jipi
ISSN: 2540-8984
Vol. 8, No. 3, September 2023, Pp. 736-743

# *N*-DAYS TOURIST ROUTE RECOMMENDER SYSTEM IN YOGYAKARTA USING GENETIC ALGORITHM METHOD

**Muhammad Ridha Anshari[1)], Z. K. A Baizal*[2)]**

1. Informatics, School Of Computing, Telkom University, Bandung, Indonesia
2. Informatics, School Of Computing, Telkom University, Bandung, Indonesia

**Article Info**

**Keywords:** Genetic Algorithm; Multi-Attribute Utility Theory; Recommender System; Tour Planning

**Article history:**
Received 3 March 2023
Revised 17 March 2023
Accepted 31 March 2023
Available online 1 September 2023

**DOI :**
https://doi.org/10.29100/jipi.v8i3.3893

* Corresponding author.
Z. K. A Baizal
E-mail address:
baizal@telkomuniversity.ac.id

**ABSTRACT**

Tourism is one of the proven solutions for the Indonesian economy. Tourism in certain regions, such as Yogyakarta, can significantly affect the region's economic development, including creating new jobs, creating new business opportunities, and increasing regional income. However, for tourists from outside Yogyakarta, it requires planning a tour before traveling in Yogyakarta, especially if he wants to spend several days on a tour. Many previous studies have developed systems that can recommend tourist routes, but not within a few days of tourist visits. In this study, we propose the use of Genetic Algorithm (GA) for automatically generating optimal travel itinerary for some days visit (*n*-days tour route). We develop the recommender system by combining GA and the concept of Multi-Attribute Utility Theory (MAUT). This MAUT used for accommodating user needs based some criteria such as rating, cost, and time. Based on our experimental results, GA is optimal in terms of execution time and number of attractions visited in *n*-days visit. The average execution time obtained is 59.62%, and the average number of attractions visited obtained is 45.95%. These results show that this method can generate tourist routes efficiently.

## I. INTRODUCTION

TOURISM is an activity that involves moving locations, individuals or groups to enjoy the natural beauty. In addition, tourism can also be a solution to support the economy of a region [1]. Yogyakarta is one of the cities with the highest number of domestic tourist visits, with 24,990,414 people [2]. However, the number of tourist attractions in Yogyakarta needs to be clarified. Therefore, a system is needed to provide recommendations for the right tour.

The approach used for the route planning problem is using the Traveling Salesman Problem (TSP). TSP is an optimization problem where a person must visit a certain number of cities efficiently, considering transportation costs and the time required to visit each city [3]. To solve TSP, several methods can be used, including using algorithms such as Tabu Search [4], Simulated Annealing [5], Ant Colony [6], Genetic Algorithm [7], Artificial Bee Colony [8], and others.

In previous research, Baizal et al. [9], [10] made a tourist route recommender system using the Tabu Search algorithm and a tourist route recommender system based on user preferences using the Simulated Annealing algorithm.

A genetic algorithm (GA) is used to handle optimization search problems. It is based on concepts like how natural selection and evolution operate in biology [11]. GA operates by regulating a population of individuals, each representing a different solution to the problem. The ideal course of action will be determined using each individual's fitness value, which reflects their qualities [12].

In previous works, there have been several methods used to develop route recommender system using GA. Wibowo [13] applied the GA to create a tourist route planning by visiting tourist attractions and restaurants promptly. Yoga [14] built a transportation management system by applying the GA. Liu Xin [15] created a logistics distribution route optimization solution using the GA method. Phatpicha [16] create Travel Itinerary Recommendation with POI using GA method. However, these studies developed the system that recommends optimal route in one day trip.

With so many tourist attractions in Yogyakarta, it might be difficult for visitors, especially from outside Yogyakarta, to determine the best tourist route. Recently, many tourists want to spend some days for tourism. In earlier study, many studies developed the recommender system for recommending optimal routes for one day route.

JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)
Journal homepage: https://jurnal.stkippgritulungagung.ac.id/index.php/jipi
ISSN: 2540-8984
Vol. 8, No. 3, September 2023, Pp. 736-743

In this study, we proposed the use of GA for developing *n*-days tourist route. This system creates travel routes in a few days (*n*-days) and personalizes tour route by considering three criteria related to the needs of tourists: the number of tourist attractions to visit, the popularity of tourist attractions, and travel cost. We consider the travel route determination problem as a multicriteria problem and use Multi-Attribute Utility Theory (MAUT) to solve it. MAUT is a theory often used to evaluate products based on the weight and utility values of multiple criteria or attributes [17]. MAUT is often used in decision-making processes that require the evaluation of complex criteria or attributes that cannot be measured directly [18].

## II. RESEARCH METHOD

The system created is a Yogyakarta tourist route recommender system. The system is built using GA and MAUT. There are many things to think about when planning tourist routes, like how busy the roads are based on Google Maps data, how long people stay at each tourist spot based on Google data, and when each tourist spot opens and closes.

### *A. Data Collecting*

The collection of data is crucial to the development of an effective tourist route recommender system. The information gathered about tourist spots in the Yogyakarta area includes their names, types, addresses, contact information, opening hours, coordinates (latitude and longitude), and distances. This information is gathered using a number of public APIs, such as Google Maps and SerpAPI.

### *B. System Design*

The use of GA is implemented in the development of this tourist route recommender system. The case study of the city of Yogyakarta is a demonstration example of the implementation of this system to find tourist routes. The design of this system includes Genetic Algorithm implementation and MAUT calculation, which is illustrated in Fig. 1

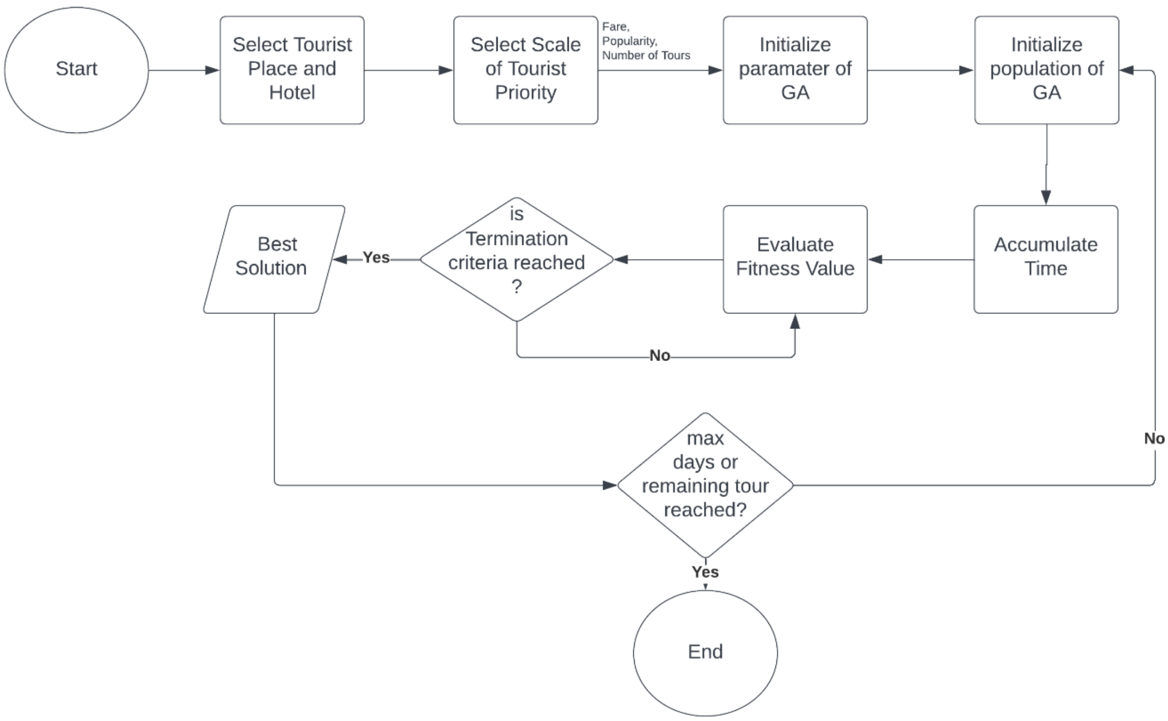


Fig. 1. Flow Design of GA

### *C. Selection of Tourist Places and Hotels*

In the first stage, the tourists will choose the hotel as the starting and ending location in the route-finding process using the Genetic Algorithm. However, this research will include all tourist attractions in the available dataset. Table 1 shows the example of data.

JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)
Journal homepage: https://jurnal.stkippgritulungagung.ac.id/index.php/jipi
ISSN: 2540-8984
Vol. 8, No. 3, September 2023, Pp. 736-743

TABLE I
THE EXAMPLE OF DATA

| ID | Cost | Rating | Spend Time(s) | Operating Time |
|---|---|---|---|---|
| 0 | 25000 | 5,0 | 3200 | 08:00 - 12:00 |
| 1 | 15000 | 4,2 | 1800 | 08:00 - 18:00 |
| 2 | 0 | 4,5 | 2200 | 10:00 - 18:00 |
| 3 | 20000 | 4,6 | 2400 | 09:00 - 16:00 |

### D. *Select Scale of Tourist Priority (Budget, Popularity, Number of tours)*

In the second stage, tourists will select a scale of importance for tourist trips based on three criteria: cost, popularity, and amount of tours. Each attribute's weight is a value between 0 and 1. At this point, the calculation is performed using the MAUT approach. The MAUT calculation seeks to establish the appropriate value, which will subsequently be utilized to calculate the fitness value.

### E. *Initialization Parameter of GA*

In the third stage, we run parameter initialization for the GA. Parameter initialization is useful for setting initial values used in the next process. The initialized parameters include population (set of routes), maximum days (*MAX_DAY*), population size (*POP_AMOUNT*), mutation probability (*PM*), and chromosome probability (*PC*).

### F. *Initialization GA Population*

In the fourth stage, the GA population consists of several individuals, each representing a possible route for a traveler to visit a set of cities. Each individual in the population consists of a set of genes that store information about the sequence of cities that the traveler should visit.

### G. *Time Accumulation*

Time Accumulation is calculated based on the number of days chosen by the traveler, assuming that the tour starts and ends at the hotel chosen by the traveler. The visit time is limited to 08.00–20.00. To determine the travel route per day, the system verifies the hotel or tourist attraction's opening and closing times before a visit. If the tourist attractions are closed, tourists will be sent to other open tourist attractions. This stage's result is a list of tourist sites accessible within the specified time limits, which is used to compute the fitness value.

### H. *Calculate Fitness Value*

The fitness value is used to figure out how good the solution (tour) is. In this study, the MAUT method is used to figure out the fitness value because it considers three factors: the number of tourist attractions to see, the popularity of tourist attractions, and travel costs. The formula used to compute the fitness value is (1).

$$F(t,r,b) = \frac{w1 \times t + w2 \times r + w3 \times b}{3} \tag{1}$$

In the equation, function $f(t,r,b)$ represent the entire performance evaluation result, where $w1$, $w2$, and $w3$ are the weights of each evaluation criterion, and $t$, $r$, and $b$ are the values of each criterion.

### I. *Evaluate Fitness Value*

After the fitness value of the obtained solution is calculated, we must first check whether the new solution obtained is better than the previous solution. In GA, selecting the best solution is done using selection techniques [19]. In the GA that is built, tournament selection is used. This method determines the best individual from a group of randomly selected individuals. Then the best individual is chosen as a parent for the next generation [20].

### J. *Termination Condition in GA*

In this stage, the conclusion of the running GA is decided. This is crucial in determining when to stop running the GA to get the best results. This research sets the halting criteria as the maximal number of iterations. After the maximal number of iterations is acquired, the best position to be in for a single day is found.

### K. *Cutting the Tour*

Once the best position for a day is obtained, the remaining tour sets that do not fall under that position will be moved to the next day. This process will be carried out until the maximum number of days is reached, or there are no more tour sets left. The tour solution is computed by adding the travel time from one attraction spot or hotel to another with the time tourists spend at each place. The system will examine each route's accumulated time and

JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)
Journal homepage: https://jurnal.stkippgritulungagung.ac.id/index.php/jipi
ISSN: 2540-8984
Vol. 8, No. 3, September 2023, Pp. 736-743

opening and closing hours for each route. If the seventh node exceeds the specified time of 08.00 - 20.00 and when going to the eighth node, the tourist spot is already closed, then the node will be used for the next day, then proceed to the next node until completion. This process ensures that all tours are on schedule and that all are included. The illustration can be seen in Fig. 2.

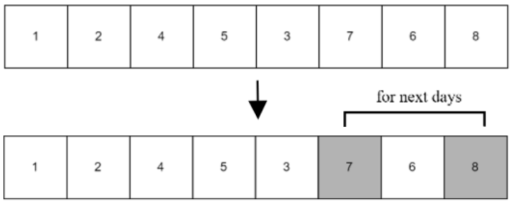

Fig. 2 The Illustration of Cutting the Tour

## III. RESULT AND DISCUSSIONS

This section discussed the sample calculation of GA, testing scenario, parameter test, performance test, and visualization of tourist routes.

### *A. Sample Calculation of GA*

A sample of how to manually calculate the GA process from the formation of individuals to get the best final result in 1 iteration, with an example of each individual containing the coordinates of tourist attractions, the cost of tourist attractions, popular tourist attractions, and the operating time of tourist attractions. Here is the process:

1) Generate Individuals
   Create individuals that have the coordinates of tourist attractions, the cost of tourist attractions, the operating time of tourist attractions, and the spending time of tourist attractions as follows:
   Individual 1: [x1, y1, 50000, "09:00-17:00", 4]
   Individual 2: [x2, y2, 75000, "10:00-18:00", 3]
   Individual 3: [x3, y3, 100000, "11:00-19:00", 2]
   Individual 4: [x4, y4, 125000, "12:00-20:00", 1]
2) Selection
   Tournament 1:
   Select two individuals randomly from the four existing individuals. Compare the fitness value. The better fitness value will pass and proceed to the next stage.
   Tournament 2:
   Choose two individuals randomly from four existing individuals. Compare the fitness value. The better fitness value will qualify and proceed to the next stage.
3) Crossover
   Use the uniform crossover method by taking one gene from the individual that passes tournament one and combining it with the gene from the individual that passes tournament 2. For example, the new individual has the coordinates [x1, y4, 125000, "09:00-17:00", 1].
4) Mutation
   Randomly select a new individual generated from the crossover process and change one of its genes. For example, change the operating time to "09:00-18:00".
5) Result
   The best result obtained is the new individual resulting from the crossover and mutation process, with coordinates [x1, y4, 125000, "09:00-18:00", 1].

### *B. Testing Scenario*

At this point, the system is tested to see how well it was put together. Two tests are conducted:

1) To produce optimal results, testing is carried out on the system parameters that have been made by combining several parameter values in the GA, such as, *POP_AMOUNT*, *PC*, and *PM*.
2) To compare the performance of the algorithm used in the same case to that of another heuristic algorithm, the Memetic Algorithm (MA), testing was done using the TSP findings for Execution Time (*Exe_Time*), number of nodes (*Attractions*), and length of time in one day.

JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)
Journal homepage: https://jurnal.stkippgritulungagung.ac.id/index.php/jipi
ISSN: 2540-8984
Vol. 8, No. 3, September 2023, Pp. 736-743

### *C. Parameter Test of GA*

In the parameter test phase, we examined parameters such as *POP_AMOUNT*, *PC*, and *PM* that affect the best search results in the method used. We ran 30 epochs, mixed the chosen parameter sets with *POP_ AMOUNT* = [30, 50, 100], *PC* = [0,3, 0,4, 0,5, 0,6, 0,9], and *PM* = [0,1, 0,4, 0,6, 0,8, 0,10, 0,05], and then evaluated each combination five times. The best parameters were found by making a scoring system that considered the average of *Attractions*, *Exe_Time*, and *N_Days*. The calculation of the *score* is defined by (2).

$$score = \frac{NN + NT + ND}{3} \tag{2}$$

In (2), *NN* represents normalized result of the number of *Attractions*, *NT* represents normalized result of the sum of *Exe_Time*, and *ND* represents normalized result of the number of *N_Days*.

In the parameter test, 90 combinations of parameter sets were obtained. The combinations of tested parameter sets are shown in Table 2. Then, we used an optimal configuration of parameters, such as *POP_AMOUNT* = 30, *PC* = 0,5, and *PM* = 0,8.

TABLE II
BEST 5 PARAMETERS OF GA

| POP_AMOUNT | PC | PM | Score |
|---|---|---|---|
| 30 | 0,5 | 0,8 | 0,782 |
| 30 | 0,9 | 0,1 | 0,679 |
| 30 | 0,3 | 0,1 | 0,670 |
| 50 | 0,4 | 0,4 | 0,667 |
| 30 | 0,4 | 0,1 | 0,661 |

Wibowo et al. [13] used GA with parameters of 46 for *POP_AMOUNT*, 0,72 for *PC*, and 0,27 for *PM* in previous research. We tested these GA parameters and obtained a score of 0,5. Then, we also tested the best parameters show in Table 2 and obtained a score of 0,782.

### *D. Performance Test*

Performance is evaluated by comparing the suggested method to the MA method. Table 3 shows that the first test is a multi-criteria test with equal weights. As a result, the average number of attractions is the same between GA and MA, which is 45,7, as shown in Fig. 4. However, the execution time of GA is faster, while MA is slower, as shown in Fig. 3. Both scores have almost the same value as shown in Fig. 5.

TABLE III
PERFORMANCE OF GA AND MA WIH MULTICRITERIA WEIGHT (T = 1, R = 1, B = 1)

| Input Attractions | GA | | | | MA | | | |
|---|---|---|---|---|---|---|---|---|
| | Attractions | N_Days | Exe_Time | Score | Attractions | N_Days | Exe_Time | Score |
| 5 | 4 | 1 | 0,495 | 0,934 | 4 | 1 | 1,607 | 0,934 |
| 15 | 14 | 2 | 1,044 | 0,924 | 14 | 2 | 6,004 | 0,925 |
| 30 | 28,6 | 4,4 | 4,114 | 0,887 | 28,6 | 4,4 | 16,621 | 0,882 |
| 45 | 43,6 | 7,4 | 7,631 | 0,859 | 43,4 | 7,4 | 48,149 | 0,86 |
| 60 | 58,6 | 10,6 | 12,143 | 0,846 | 58,4 | 10,4 | 71,226 | 0,845 |
| 75 | 74 | 13,6 | 18,429 | 0,829 | 74 | 14 | 105,508 | 0,832 |
| 100 | 97,6 | 17,2 | 33,767 | 0,84 | 97,8 | 18 | 790,912 | 0,841 |
| Mean | **45,6** | **8,028** | **11,089** | **0,876** | 45,771 | 8,171 | 1448,578 | 0,874 |

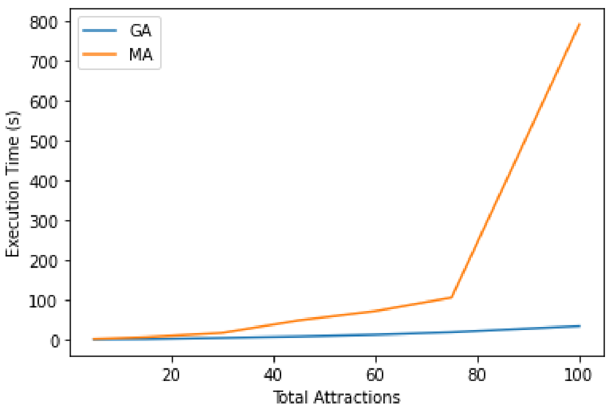

Fig. 3 Execution Time vs Total Attraction of GA and MA

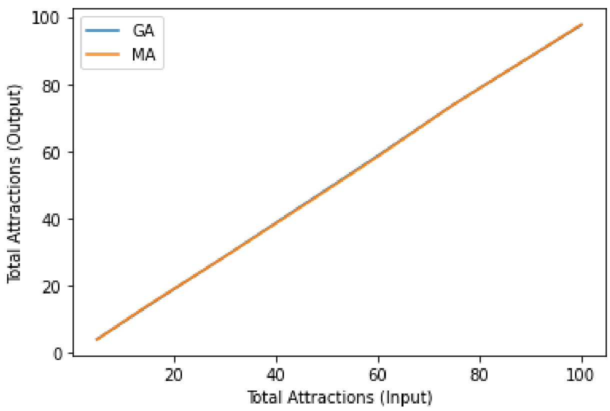

Fig. 4 Total Attraction (Input) vs Total Attraction (Output) of GA and MA

JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)
Journal homepage: https://jurnal.stkippgritulungagung.ac.id/index.php/jipi
ISSN: 2540-8984
Vol. 8, No. 3, September 2023, Pp. 736-743

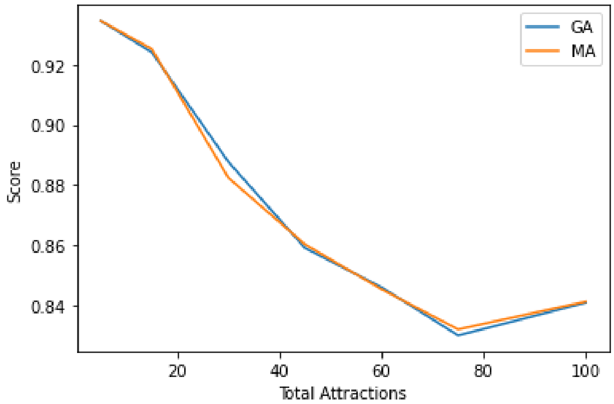


Fig. 3 Score vs Total Attractions of GA and MA

We tested for single criteria, i.e., there are three criteria for this test: number of tourist attractions, popular tourist attractions, and cost. Table 4 shows the performance results of GA and MA for the number of days criterion, Table 5 shows the performance results of GA and MA for the popular criterion, and Table 6 shows the performance results of GA and MA for the cost criterion. The results of these tests show that the GA outperforms the MA in terms of the number of attractions and execution time.

TABLE IV
PERFORMANCE OF GA AND MA WITH SINGLECRITERIA WEIGHT (T = 1, R = 0, B = 0)

| Input Attractions | GA | | | | MA | | | |
|---|---|---|---|---|---|---|---|---|
| | Attractions | N_Days | Exe_Time | Score | Attractions | N_Days | Exe_Time | Score |
| 5 | 4 | 1 | 0,630 | 0,289 | 4 | 1 | 5,139 | 0,289 |
| 15 | 14,8 | 2 | 1,382 | 0,276 | 15 | 2 | 24,989 | 0,286 |
| 30 | 29,6 | 5 | 4,263 | 0,253 | 29 | 4,4 | 87,766 | 0,258 |
| 45 | 43,6 | 8 | 8,846 | 0,248 | 43,2 | 6,8 | 131,577 | 0,250 |
| 0 | 58,6 | 10,0 | 35,162 | 0,244 | 59 | 10,2 | 226,544 | 0,242 |
| 75 | 74,6 | 13,4 | 51,914 | 0,231 | 72,2 | 12,6 | 342,480 | 0,241 |
| 100 | 98,8 | 18,0 | 68,749 | 0,233 | 97,8 | 17,0 | 2809,538 | 0,243 |
| Mean | **46,285** | 8,200 | **24,421** | 0,253 | 45,742 | **7,714** | 518,290 | **0,258** |

TABLE V
PERFORMANCE OF GA AND MA WITH SINGLECRITERIA WEIGHT (T = 0, R = 1, B = 0)

| Input Attractions | GA | | | | MA | | | |
|---|---|---|---|---|---|---|---|---|
| | Attractions | N_Days | Exe_Time | Score | Attractions | N_Days | Exe_Time | Score |
| 5 | 4 | 1 | 0,420 | 0,311 | 4 | 1 | 2,115 | 0,311 |
| 15 | 13,2 | 2,2 | 1,814 | 0,309 | 13 | 2 | 9,822 | 0,310 |
| 30 | 29,6 | 6 | 3,951 | 0,301 | 29,2 | 5,2 | 26,196 | 0,301 |
| 45 | 4486 | 9,4 | 9,230 | 0,301 | 44,8 | 8,8 | 66,317 | 0,301 |
| 60 | 59,6 | 12,8 | 11,002 | 0,302 | 59,6 | 12,2 | 107,181 | 0,302 |
| 75 | 74,6 | 16 | 16,551 | 0,303 | 74,4 | 16,0 | 175,068 | 0,303 |
| 100 | 98,4 | 20 | 26,655 | 0,302 | 98,6 | 19,2 | 246,953 | 0,302 |
| Mean | **46,314** | 9,628 | **9,946** | 0,304 | 45,771 | **9,200** | 90,552 | 0,304 |

TABLE VI
PERFORMANCE OF GA AND MA WITH SINGLECRITERIA WEIGHT (T = 0, R = 0, B = 1)

| Input Attractions | GA | | | | MA | | | |
|---|---|---|---|---|---|---|---|---|
| | Attractions | N_Days | Exe_Time | Score | Attractions | N_Days | Exe_Time | Score |
| 5 | 4 | 1 | 0.364 | 0.333 | 4 | 1 | 3.075 | 0.333 |
| 15 | 14 | 2 | 1.624 | 0.330 | 14 | 2 | 12.890 | 0.330 |
| 30 | 28.6 | 5.0 | 5.009 | 0.317 | 28.6 | 5.0 | 49.218 | 0.316 |
| 45 | 43.6 | 8.6 | 10.274 | 0.311 | 43.4 | 8.4 | 73.195 | 0.312 |
| 60 | 58.6 | 12.4 | 17.629 | 0.302 | 58.4 | 12.2 | 98.282 | 0.309 |
| 75 | 74 | 15.0 | 40.409 | 0.298 | 74 | 15.4 | 149.051 | 0.296 |
| 100 | 97.6 | 19.2 | 33.876 | 0.301 | 97.8 | 19.2 | 222.847 | 0.300 |
| Mean | 45.6 | 9.028 | **14.169** | 0.313 | **45.771** | 9.028 | 86.937 | **0.314** |

## *E. Tourism Route Visualization*

The results of the travel route scheduling system using the GA can be seen in Fig. 6. The visualization displays a map containing GA-recommended trip places.

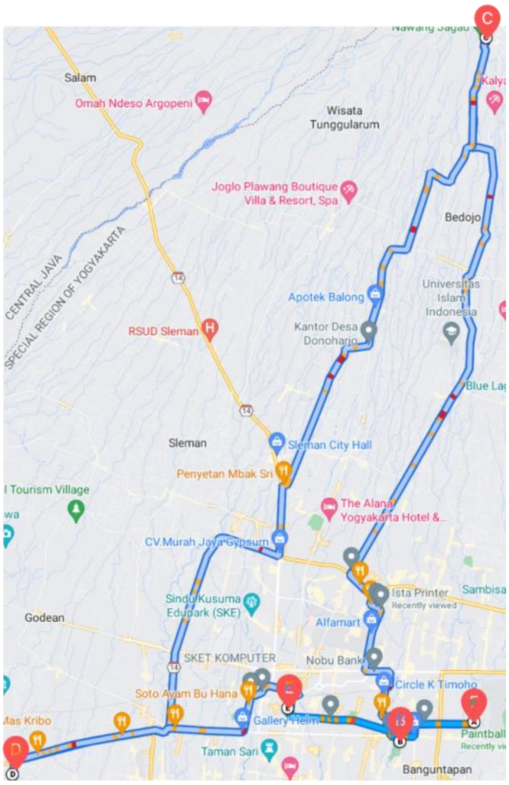


Fig. 4 The Route Visualization

## IV. CONCLUSION

By using GA and MAUT, we made an optimal recommendation system that provides route suggestions based on the number of days the user needs (N-days) and user preferences. Google Maps and SerpAPI datasets were used. GA is used to find the optimal route, considering the operating hours at tourist attractions. Then, MAUT is used to compare how well different solutions work based on factors such as popularity, cost, and travel time. We found the ideal parameters after testing ninety different parameter combinations. Based on our experimental findings for GA parameters, we determined that *POP_AMOUNT* = 30, *PC* = 0.5, and *PM* = 0.08, with a total score of 0.782. These variables were used in experiments. Experiments were carried out by comparing this method with the MA method. Although the scores for GA and MA algorithms are almost the same, GA has a shorter average computation time and more objects found. The average computation time is 59.62%, and the average number of attractions found is 45.95%. These results show that the GA approach can build an optimal itinerary quickly.

## REFERENCES

[1] M. Tenemaza, S. Luján-Mora, A. de Antonio, and J. Ramírez, “Improving Itinerary Recommendations for Tourists Through Metaheuristic Algorithms: An Optimization Proposal,” *IEEE Access*, vol. 8, pp. 79003–79023, 2020, doi: 10.1109/ACCESS.2020.2990348.

[2] A. Wicaksono, “New normal pariwisata yogyakarta,” *Kepariwisataan: Jurnal Ilmiah*, vol. 14, no. 3, pp. 139–150, 2020.

[3] N. Hanafiah, I. Wijaya, S. Xavier, C. G. Young, D. Adrianto, and M. Shodiq, “Itinerary recommendation generation using enhanced simulated annealing algorithm,” *Procedia Comput Sci*, vol. 157, pp. 605–612, 2019.

[4] R. W. Dewantoro, P. Sihombing, and others, “The combination of ant colony optimization (ACO) and tabu search (TS) algorithm to solve the traveling salesman problem (TSP),” in *2019 3rd International Conference on Electrical, Telecommunication and Computer Engineering (ELTICOM)*, 2019, pp. 160–164.

[5] L. Wang, R. Cai, M. Lin, and Y. Zhong, “Enhanced list-based simulated annealing algorithm for large-scale traveling salesman problem,” *IEEE Access*, vol. 7, pp. 144366–144380, 2019.

[6] Y. Wang and Z. Han, “Ant colony optimization for traveling salesman problem based on parameters optimization,” *Appl Soft Comput*, vol. 107, p. 107439, 2021.

[7] P. M. Hariyadi, P. T. Nguyen, I. Iswanto, and D. Sudrajat, “Traveling salesman problem solution using genetic algorithm,” *Journal of Critical Reviews*, vol. 7, no. 1, pp. 56–61, 2020.

[8] L. S. Hasan, “Artificial Bee Colony Algorithm and Bat Algorithm for Solving Travel Salesman Problem,” *Webology*, vol. 19, no. 1, pp. 4185–4193, Jan. 2022, doi: 10.14704/web/v19i1/web19276.

[9] M. Anranur Uwaisy, Z. K. A. Baizal, and M. Yusza Reditya, “Recommendation of scheduling tourism routes using tabu search method (case study bandung),” in *Procedia Computer Science*, 2019, vol. 157, pp. 150–159. doi: 10.1016/j.procs.2019.08.152.

[10] Z. K. A. Baizal, K. M. Lhaksmana, A. A. Rahmawati, M. Kirom, and Z. Mubarok, “Travel route scheduling based on user’s preferences using simulated annealing,” *International Journal of Electrical and Computer Engineering*, vol. 9, no. 2, pp. 1275–1287, 2019, doi: 10.11591/ijpeds.v9i2.pp1275-1287.

[11] V. Chahar, S. Katoch, and S. Chauhan, “A Review on Genetic Algorithm: Past, Present, and Future,” *Multimed Tools Appl*, vol. 80, Jan. 2021, doi: 10.1007/s11042-020-10139-6.

[12] R. T. Prasetio, “Genetic Algorithm to Optimize k-Nearest Neighbor Parameter for Benchmarked Medical Datasets Classification,” *Jurnal Online Informatika*, vol. 5, no. 2, pp. 153–160, 2020.

[13] B. S. Wibowo and M. Handayani, "A Genetic Algorithm for Generating Travel Itinerary Recommendation with Restaurant Selection," in *2018 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM)*, 2018, pp. 427–431. doi: 10.1109/IEEM.2018.8607677.
[14] I. W. A. K. Yoga *et al.*, "Advanced Traveler Information Systems: Itinerary Optimisation Using Orienteering Problem Model and Genetic Algorithm," in *2018 International Conference on Information Technology Systems and Innovation (ICITSI)*, 2018, pp. 454–459. doi: 10.1109/ICITSI.2018.8695952.
[15] L. Xin, P. Xu, and G. Manyi, "Logistics distribution route optimization based on genetic algorithm," *Comput Intell Neurosci*, vol. 2022, 2022.
[16] P. Yochum, L. Chang, G. Tianlong, M. Zhu, and H. Chen, "A Genetic Algorithm for Travel Itinerary Recommendation with Mandatory Points-of-Interest," 2020, pp. 133–145. doi: 10.1007/978-3-030-46931-3_13.
[17] Sunardi, Rusydi Umar, and D. Sahara, "Best Employee Decision Using Multi Attribute Utility Theory Method," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 6, no. 6, pp. 945–951, Dec. 2022, doi: 10.29207/resti.v6i6.4318.
[18] H. J. Jun, J. H. Kim, D. Y. Rhee, and S. W. Chang, "'SeoulHouse2Vec': An embedding-based collaborative filtering housing recommender system for analyzing housing preference," *Sustainability*, vol. 12, no. 17, p. 6964, 2020.
[19] S. Mirjalili, "Genetic Algorithm," in *Evolutionary Algorithms and Neural Networks: Theory and Applications*, Cham: Springer International Publishing, 2019, pp. 43–55. doi: 10.1007/978-3-319-93025-1_4.
[20] S. Prayudani, A. Hizriadi, E. Nababan, and S. Suwilo, "Analysis Effect of Tournament Selection on Genetic Algorithm Performance in Traveling Salesman Problem (TSP)," *J Phys Conf Ser*, vol. 1566, p. 12131, Jan. 2020, doi: 10.1088/1742-6596/1566/1/012131.