© Universiti Tun Hussein Onn Malaysia Publisher's Office

IJIE

Journal homepage: http://penerbit.uthm.edu.my/ojs/index.php/ijie
ISSN : 2229-838X e-ISSN : 2600-7916

The International Journal of Integrated Engineering

Analysis Simple Additive Weighting and Genetic Algorithm for Traffic Management System

Abdul Aziz[1], Surya Michrandi Nasution[2], Casi Setianingsih[1*]

[1]Telkom University,
School of Electrical Engineering, Bandung, 40257, INDONESIA

[2]Telkom University,
School of Electrical Engineering, Bandung, 40257, INDONESIA

*Corresponding Author

DOI: https://doi.org/10.30880/ijie.2020.12.07.012
Received 30 March 2020; Accepted 9 August 2020; Available online 30 August 2020

Abstract: Bandung Tourism is currently developing rapidly, where every year the number of tourist attractions has increased. The convenience provided by today's technology such as Google Maps is still lacking in helping tourists. Until now, software that is useful for determining the route by selecting tourist attractions is still small. The problem of tourists in making a tour include traffic jams, distances and tourist attractions to be visited. Application development to help find the best route is very much needed by tourists. The best route search optimization can be done using Genetic Algorithms. Genetic Algorithms are often used in determining the route because based on previous research it produces optimal results. Weighting for a path can be done using the Simple Additive Weighting Algorithm. In this study optimization of route selection is done in the hope that it can provide solutions to tourists in route selection.

Keywords: Genetic Algorithm, Simple Additive Weighting Algorithm, Distance, Traffic Congestion, number of tourist attractions.

1. Introduction

Bandung city has many tourist attractions that are visited by domestic and foreign tourists. Tourism visited by tourists in the city of Bandung includes nature reserves, maritime tours, plantations, agriculture and cultural tourism. Bandung city revenue in the tourism sector from year to year continues to increase, for 2016 alone, the number of domestic tourists visiting Bandung amounted to 4,827,589 people, while for foreign tourists amounted to 173,036 people [1]. For the budget given by the government in the tourism sector in 2016 alone Rp. 5 trillion. Increased visitors and incomes cause tourism sites in the city of Bandung itself to be increasingly in line with technological developments. To facilitate tourists in finding the path to be taken and choosing a place to visit one of the applications that support this they can use the Google Map. Google Map itself is an application from Google based on Geographic Information System to find a location to be visited [2]. By using Google Map, we can find the location information of a place, but in Google Map itself there are still shortcomings such as optimization of which tourist attractions to visit first. Determination of the route to visit various tourist attractions in the city of Bandung can be completed with a variety of Algorithms, one of which is the Genetic Algorithm. Genetic Algorithm is one way to solve a problem because of its simplicity and ease of use in applying it to various problems. The algorithm also provides optimal results in solving problems compared to other algorithms, so that in its manufacture Genetic Algorithms are used to determine the Bandung City Tourism Route on the basis of Traveling Salesman Problems as well as to test Genetic Algorithms.

*Corresponding author: setiacasie@telkomuniversity.ac.id

2. Literature Study

2.1 Travelling Salesman Problem

The problem that arises in the TSP is how to visit a place that is marked by a node (node) on the graph from the starting point to each node that will be traversed with a minimum weight which for the minimum weight can represent various things, such as minimum costs, time, fuel, distance, minimum density and others [4][5].

2.2 Graph Teory

A graph is a collection of objects that can be represented by points, angles, or vertices connected by lines or sides. A graph can be represented by vertices and edges [6]. Usually a graph uses mathematical notation. The following is an illustration of the formula (1).

$$G=(V,E) \qquad (1)$$

V is a collection of vertices (Vertex) and E is a collection of sides (Edge) pairs of the node itself. One example of a graph is information about its connectedness between cities that are marked by points or vertices which are connected by lines or biases called edges (Edge). In a side structure (Edge) a graph can be given a value or weighting on each side[7]. This weight or value can usually be expressed in terms of the distance between these points, whereas for Vertices (Vertex) it can be used in granting heuristic values to a graph.

2.3 Euler's Circuit and Path

Euler's Path is the path through each side of the graph exactly once. If the path returns to the original node, forming a closed path (circuit), then the closed path is called the Euler circuit [6]. A graph that has an Euler circuit must have an Euler path, so the following conditions must be :

- The graphic must be connected.
- The graph has exactly two vertices of an odd degree.

2.4 Simple Additive Weigthing (SAW)

Simple Additive Weighting (SAW) is looking for the weighted sum of the performance ratings for each alternative on all attributes. The Simple Additive Weighting (SAW) method requires the process of normalization of the decision matrix (X) to a scale obtained compared with all existing alternative ratings [8].

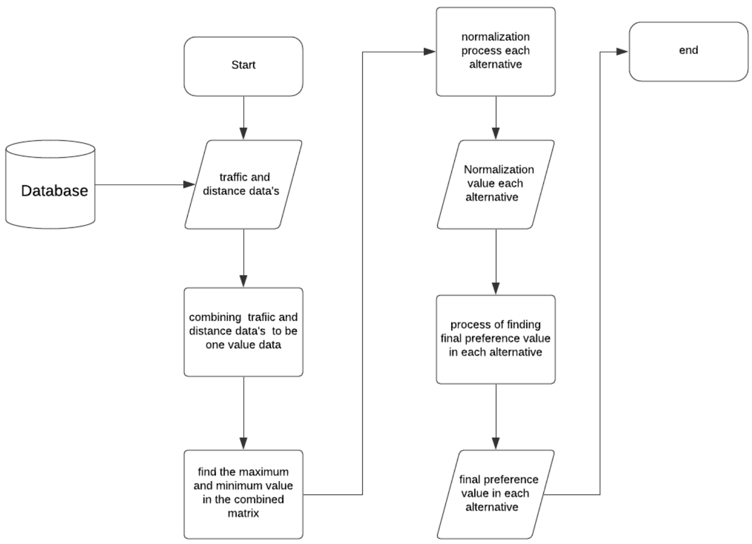

Fig. 1 - Simple Additive Weighting Process

In Figure. 1. it is explained how the Simple Additive Weighting (SAW) Method acts as a decision-making solution of the many criteria in this application such as distance, traffic jams on a highway [9].

- Retrieve Distance and traffic matrix data contained in the database.

- The values of the two matrices are combined into a combined matrix consisting of two columns, The first column is the Distance criteria, the second is traffic jams [10].
- Find the maximum and minimum values contained in each columns (criteria) of the combined matrix and determine the preference weights for each criteria. Preference weights will be used in the process of finding the final value of each alternative preference [10].
- Normalizing the existing alternatives using the maximum and minimum values obtained in the previous process.
- After the normalization process, the SAW Method will work on the process of calculating the Final Value of each alternative. This process multiplies the value obtained in the Normalization process with the Preference Weight for each alternative [8][10].
- After that, a final matrix will be obtained from the processing of the previous process. The values in this matrix will be the basic ingredients of the Genetic Algorithm process.

2.5 Genetic Algorithm (GA)

Genetic algorithm itself is an algorithm that can solve various complex problems by finding the fastest or most optimal route, genetic algorithm is an algorithm that utilizes natural selection or can also be called the evolution process [5]. One example of a problem that can be solved by genetic algorithms is getting the optimal value of problems that are likely to have many solutions. In general, the mechanism of action of the Genetic Algorithm is as follows:

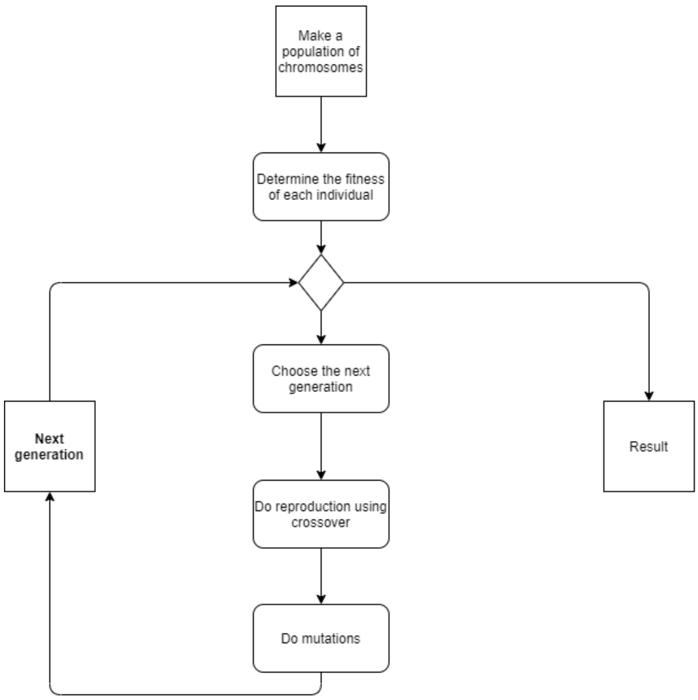

Fig. 2 - Genetic Algorithm Process

Based on Figure. 2. Illustrated how the algorithm can choose 5 sequence of tourist attractions that will be used by the user, regarding the Genetic Algorithm design diagram, there are 3 main processes:

- Selection process
  Selection is the process of determining which individuals will be selected for recombination and how offspring are formed from these selected individuals. The first step taken in the selection is the search for fitness values. The fitness value will then be used in the next selection stage [3].
- Recombination Process
  Recombination is the process of crossing two chromosomes to form a new chromosome that hopes to be better than its parent. Recombination is also known as crossover. Not all chromosomes are predetermined crossover probabilities. Crossover probability states the chance that a chromosome will experience a crossover [3].
- Mutation Process
  Mutation is the process of adding very small random values with low probability to hereditary variables. Mutase opportunities are defined as the presentation of the total number of genes in a mutated population [3]. Mutation opportunities control the number of new genes that will be raised for evaluation.

3. Design and Testing

3.1 System Overview

The system in this study makes an application optimization route for Bandung city tour using Genetic Algorithms. This system helps tourists in the city of Bandung in choosing tourist attractions based on opening hours and closing hours of tourist attractions. The following description of the system that will be created, Users can choose 5 tourist attractions, from which the 5 attractions will determine the order of tourist attractions using Genetic algorithm. Based on Figure. 3. explains the general picture of the system and processing in general. The stages of the system are as follows:

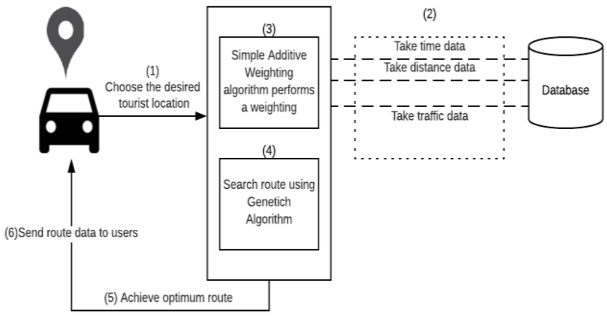

Fig. 3 - System Overview

- Tourists enter the place they want to go.
- The system will retrieve traffic data and distance data to the database based on the input of the place input by the user.
- The results of place data, distance data and traffic data will be processed using the Simple Additive Weighting Algorithm (SAW) which then results from the SAW Algorithm generate the weight value of a path.
- After getting the value of the weight of a path then the new system processes using Genetic Algorithms.
- The output of the Genetic Algorithm will produce a sequence of routes to the destination.
- The results of the Genetic Algorithm are sent again to the user's application.

3.2 Simple Additive Weighting

1. Place Initialization

In the process of the SAW algorithm using the weights of tourist attractions in the application with weights at 12 AM on Saturday following manual calculations using the SAW Algorithm, before making calculations using the SAW Algorithm first to initialization the place with numbers.

Table 1 - Initials Place

Place	Initials Place Using Numbers
Telkom University	0
Asia Afrika Museum	1
Geology Museum	2
Siliwangi Museum	3
Sri Baduga Museum	4

After initializing the place, the next step is to calculate the weight value of the chosen path based on the smallest cost of distance and congestion during the same hours and days when used by tourists.

2. Distance Matrix

Table 2 - Distance Matrix

Place	0	1	2	3	4	5
0	0	0.226	0.319	0.256	0.206	0.327
1	0.250	0	0.141	0.069	0.074	0.149
2	0.333	0.115	0	0.103	0.200	0.092
3	0.247	0.020	0.095	0	0.099	0.143
4	0.236	0.080	0.168	0.130	0	0.199
5	0.342	0.112	0.039	0.095	0.181	0

Table 2 is a matrix that contains the value of the distance of each tourist location that is connected to each other. This value is obtained based on the results of the division of tourist location distances divided by the maximum distance.

3. Traffic Matrix

Table 3 - Traffic Matrix

Place	0	1	2	3	4	5
0	0	0.375	0.48	0.36	0.497	0.48
1	0.3	0	0.4	0.3	0.6	0.3
2	0.3	0.3	0	0.3	0.3	0.3
3	0.4	0.3	0.4	0	0.15	0.3
4	0.4	0.3	0.3	0.3	0	0.3
5	0.3	0.3	0.3	0.3	0.3	0

Traffic value below 1 means that the road has moderate road conditions, if the congestion value is 1, it means the road is jammed and if the congestion value is 0 then the road is smooth.

4. Combined Matrix

Table 4 - Combined Matrix

Place	0	1	2	3	4	5
0	0	0.601	0.799	0.616	0.704	0.807
1	0.550	0	0.541	0.369	0.674	0.449
2	0.633	0.415	0	0.403	0.589	0.392
3	0.647	0.320	0.495	0	0.249	0.435
4	0.636	0.380	0.468	0.430	0	0.499
5	0.642	0.412	0.399	0.395	0.481	0

Table 4 is the result of the final weight value generated by the SAW Algorithm which is a combined weight value that is the congestion weight value plus the distance weight value which is then obtained by the final weight value.

3.2 Genetic Algorithm Processing

For the genetic algorithm calculation step we can apply it as follows using parameters like the following:

Table 5 - Parameter for Genetic Algorithm

Population	Mutation	Max Generation
5	0.5	100

In the test scenario using the first generation sample as a calculation step in the genetic algorithm.

1. Population initialization

Table 6 - Population Initialization

Population	Route(chromosome)	Fitness Value
1	0, 1, 4, 3, 5, 2	2.740
2	0, 1, 2, 5, 3, 4	2.424
3	0, 4, 5, 1, 2, 3	2.709
4	0, 3, 2, 4, 1, 5	2.584
5	0, 2, 3, 1, 5, 4	2.770

In Table 6 the process of generating population in the table consists of 5 populations, each of which has a different chromosome for the fitness value of each chromosome itself can be searched by adding up the cost value of each gene in the chromosome. For the equation itself we can use the formula:

$$Fit_{tot} = \sum F_k \qquad (2)$$

Fit_{tot} = total overall fitness value on one chromosome

F_k = individual fitness

2. Invers fitness

Table 7 - Invers Fitness

Population	Route(chromosome)	Invers Fitness Value
1	0, 1, 4, 3, 5, 2	0.364
2	0, 1, 2, 5, 3, 4	0.412
3	0, 4, 5, 1, 2, 3	0.369
4	0, 3, 2, 4, 1, 5	0.386
5	0, 2, 3, 1, 5, 4	0.360

In table 7 is the inverse fitness value, for the inverse fitness value itself can be obtained with the following equation:

$$\frac{1}{F} = \frac{1}{F_k} \quad (3)$$

$\frac{1}{F} = invers\, fitness$

3. Fitness relative

Table 8 - Fitness relative

Population	Route(chromosome)	Fitness Relative Value
1	0, 1, 4, 3, 5, 2	0.192
2	0, 1, 2, 5, 3, 4	0.217
3	0, 4, 5, 1, 2, 3	0.194
4	0, 3, 2, 4, 1, 5	0.204
5	0, 2, 3, 1, 5, 4	0.190

In Table -8 is the relative fitness value to find the fitness value we can use the following formula:

$$P_k = \frac{F_k}{Fit_{tot}} \quad (4)$$

$P_k = relative\ fitness\ of\ each\ individual$

After getting the value of F_k and the value of Fit_{tot} we can determine the relative fitness value by dividing the fitness value of each individual by the total fitness value in a population.

4. Cumulative fitness

Table 9 - Cumulative fitness

Population	Route(chromosome)	Cumulative Fitness
1	0, 1, 4, 3, 5, 2	0.192
2	0, 1, 2, 5, 3, 4	0.410
3	0, 4, 5, 1, 2, 3	0.605
4	0, 3, 2, 4, 1, 5	0.809
5	0, 2, 3, 1, 5, 4	1

In Table 9 is the value of cumulative fitness, before looking for cumulative fitness values we must first have data from the inverse fitness value, because the cumulative sum is based on the inverse fitness value.

5. Chromosome Selection with Roulette Wheel

Table 10 - Chromosome Selection

Parent	Route (chromosome)	Roulette Wheel value generated	Update Population
1	0, 1, 2, 5, 3, 4	0.353	0, 2, 1, 5, 3, 4
2	0, 2, 3, 1, 5, 4	1.178	0, 1, 3, 2, 5, 4

In Table - 10 Chromosome Selection by generating random numbers from 0 to 1 by the roulette wheel method, if the cumulative value is less than the value of the raised roulette wheel, the chromosome will be replaced with a new chromosome.

6. Crossovers and Old Populations Are Replaced with New Ones

Table 11 - Crossover

Old Chromosome	New Chromosome
0, 1, 4, 3, 5, 2	0, 2, 1, 5, 3, 4
0, 1, 2, 5, 3, 4	0, 1, 3, 2, 5, 4
0, 4, 5, 1, 2, 3	0, 4, 5, 1, 2, 3
0, 3, 2, 4, 1, 5	0, 3, 2, 4, 1, 5
0, 2, 3, 1, 5, 4	0, 1, 3, 2, 5, 4

In Table -11 it is a process of interbreeding between chromosomes on an old chromosome that results in a new chromosome.

7. Mutation

Table 12 - Mutation

Roulette Wheel value generated	Mutation	Mutation Gen
0.167	0, 2, 1, 5, 4, 3	4,3
0.098	0, 1, 3, 4, 5, 2	4,2
0.098	0, 4, 2, 1, 5, 3	2,5
0.344	0, 3, 1, 4, 5, 2	1,3

In Table -12 gene mutations are carried out on the new chromosome, for mutated genes in the mutation gene column.

8. Elitism

Table 13 - Elitism

Chromosomes After Mutation	Elitism Results	Total Fitness Value
0, 2, 1, 5, 4, 3	0, 2, 1, 5, 4, 3	2.775
0, 1, 3, 4, 5, 2	0, 3, 1, 4, 5, 2	2.469
0, 4, 2, 1, 5, 3	0, 4, 2, 1, 5, 3	2.612
0, 3, 1, 4, 5, 2	0, 3, 2, 4, 1, 5	2.584
0, 1, 3, 2, 5, 4	0, 3, 1, 4, 5, 2	2.469

In Table-13 Elitism is the process of elimination carried out by eliminating one of the chromosomes whose value is less than the value of the raised roulette wheel.

9. Result Generation 1

Table 14 - Result Generation 1

Old Chromosome	Total fitness value	New Chromosome	Total fitness value
0, 1, 4, 3, 5, 2	2.740	0, 2, 1, 5, 4, 3	2.775
0, 1, 2, 5, 3, 4	2.424	0, 3, 1, 4, 5, 2	2.469
0, 4, 5, 1, 2, 3	2.709	0, 4, 2, 1, 5, 3	2.612
0, 3, 2, 4, 1, 5	2.584	0, 3, 2, 4, 1, 5	2.584
0, 2, 3, 1, 5, 4	2.770	0, 3, 1, 4, 5, 2	2.469

After completion, Generation 1 continues with the next generation with 100 generations. With the end of generation 100 the system will search for all generations, which chromosomes have better values. The results of the system show the results after 100 generations are chromosomes 0, 3, 4, 1, 5, 2 with place:

- Telkom University
- Siliwangi Museum
- Sri Baduga Museum
- Asia Afrika Museum
- Monumen Perjuangan Rakyat Jawa Barat
- Geology Museum

Table 15 - Mutation testing

Population	Crossover	Mutation	Max Gen
100	0.5	Value to be tested	100

The test is done by observing the number of generations to get the optimal value on what generation can get the best route applied to mutation process.

Table 16 - Mutation testing value

Mutation	Generation	Best Total fitness Value
0.1	3	2.668
0.2	2	2.668
0.3	3	2.668
0.4	6	2.668
0.5	3	2.668
0.6	2	2.668
0.7	4	2.668
0.8	2	2.668
0.9	2	2.668

Next testing, crossover testing uses several parameters as follows:

Table 17 - Crossover testing

Population	Crossover	Mutation	Max Gen
100	Value to be tested	0.5	100

In Table-18 the test is done by observing the number of generations to get the optimal value on what generation can get the best route applied to crossover process.

Table 18 - Crossover testing value

Crossover	Generation	Best Total fitness Value
0.1	6	2.668
0.2	3	2.668
0.3	2	2.668
0.4	2	2.668
0.5	2	2.668
0.6	3	2.668
0.7	2	2.668
0.8	3	2.668
0.9	3	2.668

The results of testing genetic algorithms using crossovers and mutations using predetermined methods can be seen in experiments above the average optimal route obtained in the 3rd generation out of 100 generations tried in the test.

3.3 Result

Fig. 4 - Result Experiment

For the explanation Fig. 5, the vertical axis is for Generation and the X-axis is for probability, the optimal generation comparison test results can be seen that with the comparison of fixed mutations with a value of 0.5 and different crossovers, the optimal generation is generated at the probability of 0.3, 0.4 and 0.7. For comparison testing of different mutations and a fixed crossover with a value of 0.5, an optimal generation is generated at the probabilities of 0.6, 0.8 and 0.9.

4. Conclusion

Simple Additive Weighting (SAW) method and Genetic Algorithm can be designed and implemented into an android-based application for the route search process. Based on testing that has been done, the results for optimal generation crossovers with fixed mutations with a value of 0.5 are obtained with the best probability values of 0.3, 0.4, and 0.5. For the results of optimal generation mutations with fixed crossovers with a value of 0.5 obtained the best probability values of 0.2, 0.6, 0.8 and 0.9.

Acknowledgment

This research was partly support by Telkom University and the Computer Engineering major which provided opportunities, platforms, and places that greatly assisted research. The author also thanks the thesis supervisor who always accompanies and provides assistance and comments that greatly improve the manuscript.

References

[1] Bandung B. P. S. K., "bandungkota.bps.go.id," Badan Pusat Statistik, 03 02 2017. [Online]. Available: https://bandungkota.bps.go.id/statictable/2017/08/29/120/jumlah-wisatawan-mancanegara-dan-domestik-di-kota-bandung-2016.html. [Accessed 08 Apr 2019]

[2] Taufiq Hidayat, F.S. (2007). Penentuan Rute Pariwisata Kota Bandung Menggunakan Algoritma Particle Swarm Optimization, 6, 56-57

[3] Walid Ellili, M. S. A. K. (2017). Traveling Salesman Problem of Optimization based on Genetic Algorithms. Jamia Millia Islamia University

[4] Suyanto, S. M. (2011). Artificial Intelligence Searching, Reasoning, Planning & Learning. Bandung: Informatika Bandung

[5] Budiharto, W., & Suhartono D. (2014). Artificial Intelligence. Yogyakarta: Andi

[6] Munir, R. (2010). Matematika Diskrit. Bandung: Informatika Bandung

[7] Primartha, R. (2018). Beajar Machine Learning Teori dan Praktik. Bandung: Informatika Bandung

[8] Setiadi, A., & Ningsih, A.R. (2018). Penerapan Metode Simple Additive Weighting (SAW), 7, 104-109

[9] Engelbrecht, A. P. (2007). Computational Intelligence. South Africa: John willey

[10] Subagio, R. T., & Abdullah, M. T. (2017). Application of SAW (Simple Additive Weighting) Method in System. UNIKOM