JOIN (Jurnal Online Informatika)
Volume 7 No. 1 | June 2022: 97-109
DOI: 10.15575/join.v7i1.797

p-ISSN: 2527-1682
e-ISSN: 2527-9165

Performance Analysis of ACO and FA Algorithms on Parameter Variation Scenarios in Determining Alternative Routes for Cars as a Solution to Traffic Jams

Yuliant Sibaroni[1], Sri Suryani Prasetiyowati[2], Mitha Putrianty Fairuz[3], Muhammad Damar[4], Rafika Salis[5]
[1,2,3,4,5] School of Computing, Telkom University, Indonesia

Article Info

Article history:

Received August 03, 2021
Revised April 26, 2022
Accepted April 27, 2022
Published June 30, 2022

Keywords:

Alternative Route
Ant Colony Optimization
Firefly Algorithm
Parameter Optimization
Travel Time

ABSTRACT

This study proposes several alternative optimal routes on traffic-prone routes using Ant Colony Optimization (ACO) and Firefly Algorithm (FA). Two methods are classified as the metaheuristic method, which means that they can solve problems with complex optimization and will get the solution with the best results. Comparison of alternative routes generated by the two algorithms is measured based on several parameters, namely alpha and beta in determination of the best alternative route. The results obtained are that the alternative route produced by FA is superior to ACO, with an accuracy of 88%. This is also supported by the performance of the FA algorithm which is generally superior, where the resulting alternative route is shorter in distance, time, running time and there is no influence on the alpha parameter value. But in each iteration, the number of alternative routes generated is less. The contribution of this research is to provide information about the best algorithm between ACO and FA in providing the most optimal alternative route based on the fastest travel time. The recommended alternative path is a path that is sufficient for cars to pass, because the selection takes into account the size of the road capacity.

Corresponding Author:

Yuliant Sibaroni,
School of Computing,
Telkom University, Indonesia
Email: ysibaroni@gmail.com

1. INTRODUCTION

ADB stated that the average congestion index in several Asian cities is 1.24, which means it takes around 24% more time to travel during peak hours [1]. Due to traffic jams that occur every day, road users experience obstacles related to daily mobilization. Thus, when road users pass through traffic jams, they hope that a solution will emerge to avoid congestion, one of which is by finding the right alternative routes, especially for cars. In the field of information technology, such as Google Maps, there are alternative route features for various types of vehicles. The algorithm used on Google Map is the Djikstra algorithm, based on current traffic conditions and looking for the shortest path.

There are not many studies that discuss the selection of alternative routes as a solution to solve congestion problems. Some studies usually only focus on choosing the shortest path, but not based on congestion problems. Several methods to find the shortest path have been developed and combined, where each method has advantages and disadvantages. Some of the methods used include Ant Colony Optimization, Firefly Algorithm, and Genetic Algorithm.

The Ant Colony Optimization algorithm has been carried out by several researchers [2], [3], [4], [5], [6], and [7] in determining the shortest route. The ACO algorithm is taken from the behavior of the ant colony, which can find the shortest route to a food source based on the footprints of other ants. Scientifically speaking, the more ants that pass along a road, the more visible the footprints of the ants that pass through that path will be. If a path is traversed by many ants, the number will increase and cause other ants to pass through that path

[3]. In [2], alternative routes were searched using the ACO method and Simple Additive Weighting based on distance, holes, bends, and density, meanwhile [4] examines the modification of the Ant algorithm, to optimize the route of delivery of goods based on changing the speed of traffic flow in a certain section.[5] discusses the ACO-based route planning approach by considering the dynamics and predictions of the distribution of travel time. Several modifications were made by metaheuristic ACO on the ant system and the ant colony system meanwhile [6] use hybridization of the ACO algorithm in solving complex routing optimization problems. Different from previous research, [7] uses ACO in obtaining optimal scheduling [7] on RCPSP (resource constrained project scheduling problem). To solve this problem, Ant Colony Optimization algorithm is used by comparing three different pheromone structures, two of which are classical structures, while the third is a new structure. Regarding the development of online marketing, [8] applies the ACO algorithm, to solve the problem of the optimal route of food distribution and goods stacking, with a visibility measure of 0.00015. In several studies, the results obtained show that the ACO algorithm by setting static parameters and various scenarios in some cases is better than the Tabu search algorithm [9], [10].

Meanwhile, the Firefly (FA) algorithm is an algorithm developed based on the behavioral characteristics of fireflies, namely the brightness factor and the attractiveness of fireflies to each other. Brightness is determined by the value of the position function of the fireflies, while the attractiveness is determined by the light level of the fireflies. Research using FA has been carried out by several researchers. In [11], FA has been compared with several other algorithms, including the Genetic Algorithm and Swarm Optimization Algorithm. According to [12] research, the Firefly Algorithm has a good ability to do exploration and exploitation processes. Meanwhile, Research [13] applies a discrete firefly algorithm based on a two-layer coding strategy to build a milk run route optimization model, in order to minimize milk run transportation costs. The fireflies are represented by route vectors and sequence vectors. The algorithm is tested with a case study and the computational results show that the proposed algorithm outperforms the taboo search algorithm.. [14] discusses the optimization of route determination on the Travelling Salesman Problem (TSP) using a hybrid Firefly Algorithm, Meanwhile, [15] use FA through the Technique for Order Preference by Similarity to Ideal Solution (TOPSIS). The hybrid method in [14] is built from a combination of the Nearest Neighbor heuristic with the Firefly Algorithm metaheuristic method. Meanwhile [16] and [17] examined the effect of parameter optimization in scheduling problems, where the results obtained were that scheduling with parameter optimization on FA was better than without parameter optimization.

Research [18] and [19] compare the performance of the two algorithms, for route optimization and scheduling. [18] compares the performance of ACO and FA based on path length and execution time. The parameter simulation scenarios are environmental density, land area, number of available robots, and hillock plantations. Meanwhile [19] estimates the parameters of the Lotka-Volterra competition model, by comparing the performance of the swarm particle optimization method and the firefly algorithm. The result obtained is that the Average Absolute Percentage Error (MAPE) of the firefly algorithm is slightly smaller than the MAPE particle swarm optimization method. While the error variance of the firefly algorithm is lower. So it can be concluded that the firefly algorithm is better than the particle swarm optimization method.

Based on the research on ACO and FA above, it can be concluded that there is no research that compares the performance of the two algorithms directly in determining the optimal route. The performance comparison of the two algorithms is based on five aspects, namely the best alternative route chosen, the alternative route search process, the average running time, the number of alternative routes selected, and the effect of parameter value scenarios on the variation of the chosen alternative route. While the optimal alternative route is chosen based on the distance and the fastest travel time. In addition, the route determined in some of the studies above is only the shortest route, there have been no studies that have used these two algorithms in determining the optimal route using scenarios of several alpha and beta parameter values, which are the parameters for ACO and FA. For this reason, in this study, a detailed comparison of the performance of the two ACO and FA algorithms will be carried out, in determining the most optimal alternative route based on the faster time and density of the Bojongsoang-Bandung area. The expected goal is to find out the best algorithm between the two algorithms in determining the most optimal route. Another goal that is expected from this research is to find several alternative routes in the Bojongsoang-Bandung area that can be used by road users in the Bojongsoang-Bandung area to get a faster alternative route when congestion occurs.

2. METHOD

There are 4 main processes that are carried out on each algorithm in determining the optimal route, namely the process of modeling the route data graph, the parameter optimization process, the accuracy measurement process and the evaluation process. In general, the stages of the process along with the inputs and outputs of each stage carried out in our study can be made in the flowchart in Figure 1.

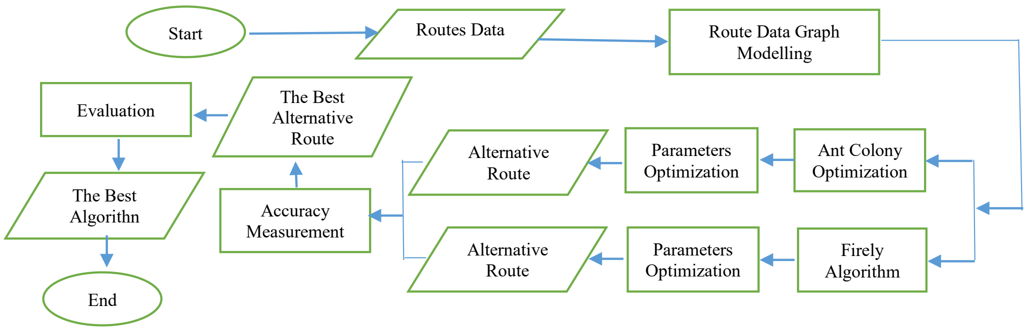


Figure 1. Process stages in determining the optimal route

2.1 Routes Data

The data in this study are primary data and secondary data. Primary data is obtained by making direct observations in the field, to get the value of the travel time and the average speed of the vehicle. Meanwhile, secondary data, namely the location and distance of the intersection, which will be used as the basis for making a graph, is obtained from google maps with a path from the Bojongsoang junction to the Buah Batu (BKR) junction.

Table 1. Node of path

Node	Junction	Node	Junction
1	Bojongsoang	15	Sukarno Hatta Batununggal
2	Cikoneng	16	Sukapura Yogya
3	Telkom	17	Sukapura Tol
4	Ciganitri	18	Sukapura Batununggal
5	Yogya	19	Cikoneng
6	Jembatan Tol	20	Ciganitri Utara
7	Sukarno Hatta	21	Ciganitri Mukti
8	Cijagra	22	Raya Ciganitri
9	BKR	23	Terusan Logam
10	Telkom Sukapura	24	Margacinta
11	Mengger Hilir Sukapura	25	Ibrahim Adjie
12	Mengger Hilir	26	Kiaracondong
13	Mengger Hilir Batununggal	27	Ciganitri Tengah
14	Batununggal		

2.2 Route Modeling

The path modeling developed in this study includes the main route which is a congestion problem and a proposed model of all possible alternative routes. The main and alternative routes developed are described in the graph in Figure 1, Figure 2 and Appendix A.

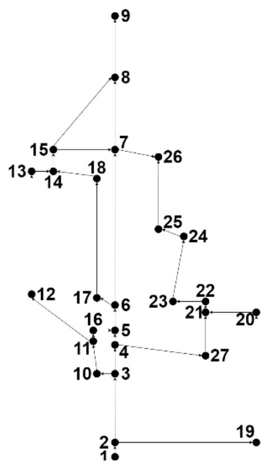


Figure 2. Graph representation of the main route

2.3 Implementation of Algorithms

This study proposes several alternative optimal routes on traffic-prone routes using Ant Colony Optimization (ACO) and Firefly Algorithm (FA). Two methods are the metaheuristic method, which means that they can solve optimization problems and will get the best solution. The implementation of the two algorithms is described in each of the following sub-chapters.

2.3.1 Implementation of Ant Colony Optimization on Alternative Path Search

There are 10 stages in the implementation of the Ant Colony Optimization algorithm in finding alternative paths. These stages can be seen in Figure 2.

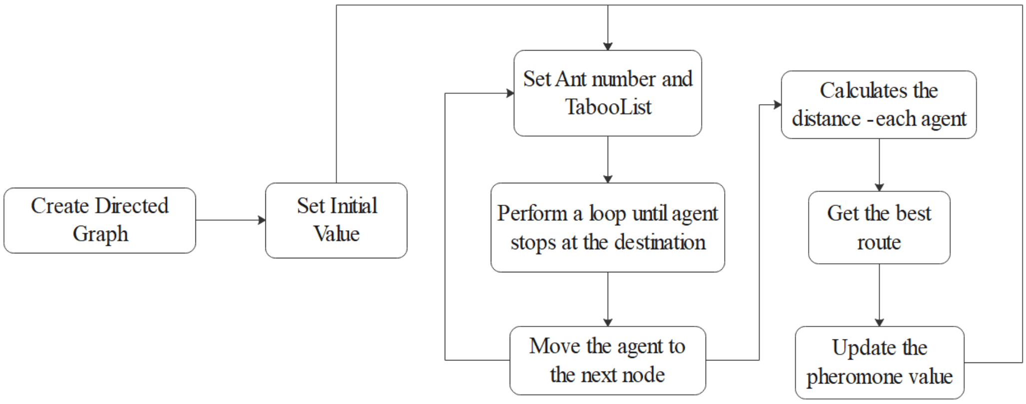


Figure 2. The implementation of the Ant Colony Optimization algorithm

Process of creating directed graph including the process of determination of the starting point and the destination point. The initial values are: iteration, number of agents, alpha value, beta value, velocity value (for alternative and main lines) and pheromone value [20]. The number of ants is 100 and the taboo list is set by 1. The probability of the movement of each agent from the initial node to the next node is [3], [20] can be seen in (1). The calculation of time for each agent is obtained by using formula (2). The best route obtained from the best short time path. Update the pheromone value performed at the end of the algorithm iteration when all agents have completed a tour. The formula of the pheromone value can be seen in equation (3).

The probability of the movement of each agent from the initial node to the next node is [3], [20]

$$p_k(i,j) = \{\frac{[\tau(i,j)]^{\alpha}.[\eta(i,j)]^{\beta}}{\sum \quad ([\tau(i,j)]^{\alpha}.[\eta(i,j)]^{\beta})} \qquad ; if\ j\ \epsilon\ allowed\ _k\ 0 \qquad ; for\ else \tag{1}$$

and $\eta(i,j) = \frac{1}{\delta}$

where

$\tau(i,j)$: pheromone value in route (i,j)

$\eta(i,j)$: visibility (invers and distances $\delta(i,j)$

δ : distance between 2 node

α : controller parameter $\tau(i,j)$, 0< α <1

β : controller parameter $\eta(i,j)$, β>0

The time calculation formula for each agent is $t = \frac{s}{v}$ (2)

where:

t: travel time agent

s: the distance traveled by the agent

v: agent speed which is assumed to be the same for each alternative route, namely 20 km / hour

The formula of the pheromone value:

$$\tau(i,j) \leftarrow (1-\rho).\tau(i,j) + \rho.\Delta\tau(i,j) \tag{3}$$

where

ρ: pheromone evaporation rate ($0 < \rho < 1$)
$\Delta\tau\ (i, j)$: change in pheromone taken based on changes in pheromone substance value

2.3.2 Implementation of Firefly Algorithm on Alternative Path Search

The implementation of the firefly algorithm is carried out in 5 stages can be seen in Figure 3

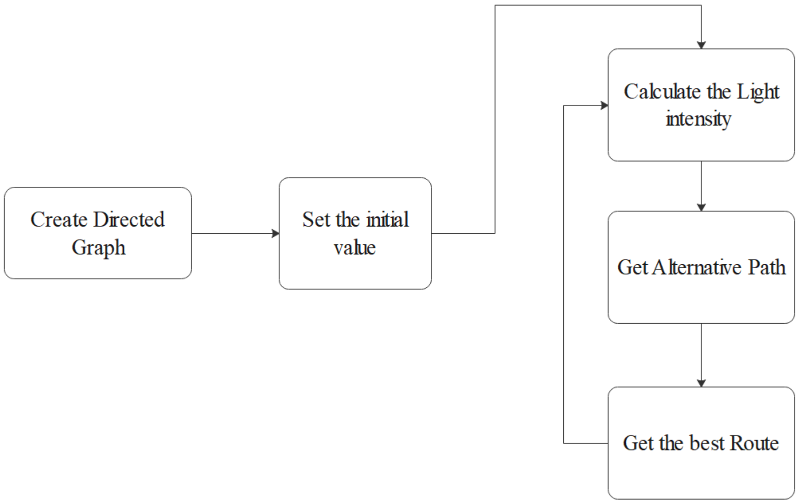


Figure 3. The implementation of the firefly algorithm

Create a directed graph based on a dataset of map coordinates, including the determination of the starting point and the destination point. The initial set values include: speed α, theta, lb, ub and the number of fireflies (random). The light intensity is calculated using formula $I = \frac{1}{t}$, and $t = \frac{s}{v}$ where

I	: Light intensity
Time(t)	: firefly travel time
Distance(s)	: the distance between 2 firefly
Speed(v)	: the preset firefly speed

If the firefly intensity value is less than the firefly intensity value, then an alternative path is found. Otherwise, it is necessary to change the position of the fireflies by detecting whether there is a path between the two fireflies. The calculation process of Light intensity will be repeated until the best route is obtained.

2.4 Testing and Accuracy Measurement

Parameters for comparing the performance of Ant Colony Optimization and Firefly Algorithm in the case of finding an alternative route starting from the Bojongsoang Intersection to Buah Batu Intersection (BKR) are based on the number of lanes and the time taken by the vehicle.

In general, to compare the treatment effect of two or more trials, analysis of variance can be used. Analysis of variance is a statistical method for detecting differences between several experimental groups, with one or more independent variables. In ANOVA, the independent variables are called factors, and the groups within each factor are called levels. The advantage of ANOVA is its ability to analyze experimental designs with several independent variables [21]. Therefore, the comparison of performance as the effect of using the alpha and beta parameter values that vary from the two algorithms. The hypotheses and analysis of variance tables are defined as follows

H_0 : All scenarios give the same response
H_1 : At least there are pairs of scenarios that give different responses

Table 2. Variance analysis table

Source of variation	Sum Square (SS)	Degree of free (Df)	Mean Square (MS)	F Value
Alpha (between groups.BG)	SSBG	k - 1	$MSBG = \frac{SSBG}{k-1}$	$\frac{MSBG}{MSWG}$
Within groups (WG)	SSWG	k (r – 1)	$MSWG = \frac{SSWG}{k\ (r-1)}$	
Total	SST	kr - 1		

Meanwhile, the evaluation process (accuracy checking) will be carried out by calculating the root mean square error (RMSE) value [22], where the algorithm with a smaller RMSE value can be said to be more accurate.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n} \quad (\hat{Z}(x_i) - Z(x_i))^2} \tag{6}$$

Where:
n : number of prediction
$\hat{Z}(x_i)$: estimation value
$Z(x_i)$: measurement value

2.5 Evaluation

Determination of the optimal alternative route based on the best travel time obtained by using several scenarios. The scenario is done by using a combination of initialization of several parameters from each algorithm. The test results will be ranked based on the travel time. The accuracy of each route which is the result of the two algorithms is compared to its Root Means Square Error (RMSE) value.

The initialization scenario of varying parameter values is used in both algorithms in order to determine the optimal alternative path. ACO and FA implementations use several scenarios, alpha and beta values.

2.5.1 Scenario of initialization of several parameters of Ant Colony Optimization

The implementation of the Ant Colony Optimization algorithm in this study, used 100 agents with 50 iterations, because from all nodes it can be identified into 9 alternative routes, where each alternative route defined there are 9 to 14 nodes, which are indeed connected. Determination of the alternative path is done when the system is looking for the travel time for the path traversed by each agent. The system will filter the travel time so that there is no recurring time, and the time selected as the time from the alternative route is the smallest of the other travel times and must be less than the travel time for the main route. The stages of implementing the ACO algorithm are described in Appendix B.

2.5.2 Scenario of initialization of several parameters of Firefly Algorithm

Meanwhile, in the implementation of the Firefly Algorithm, the number of firefly used is the same as the number of existing nodes, namely 27. One series of nodes formed by FA randomly is called the population. This population will continue to change until the nodes in the population form a best firefly. That is, in this population there are nodes that form a series of alternative paths. If the ACO system performs filtering travel time to get an alternative path, then the FA system will filter for the last firefly light intensity value that must be greater than zero. So, when the light intensity value is still zero, it will change the firefly position. The firefly position change is done in 5 repetitions. The stages of implementing the FA are described in Appendix C.

The results of scenarios using alpha and beta parameters in the Ant Colony Optimization [3],[20] and Firefly Algorithm [15], [16], [23], in the search for the best alternative routes are presented in Table 3.

Table 3. Scenarios of ACO and FA Parameters

Scenario of ACO Parameters		**Scenario of FA Parameters**
Alpha	Beta	Alpha
0.15	2	0.1
0.15	8	1
0.5	2	0.5
0.5	8	
0.85	2	
0.85	8	

3 RESULTS AND DISCUSSION

After implementing the two algorithms using alpha and beta parameters initialization scenarios as shown in the Table 3, then an analysis of the test results is carried out. In this section, the results of the selected and optimal alternative routes will be explained based on the fastest travel time. While the comparison process between scenarios and performance between the two algorithms is based on the accuracy value

3.1. Result

The implementation of the two algorithms uses a dataset that is the same, namely the distance between nodes based on the graph in Appendix A . The purpose of the two algorithms is to obtain an alternative path based on the travel time of each algorithm, then the two algorithms are compared based on the time and distance traveled.

3.1.1 Parameters Optimization of Alternative Pathway Results between Ant Colony Optimization and Firefly Algorithm

Ant colonies can find the shortest route between nests based on footprints containing pheromones. In research [24], [25] it is said that the number of ants that pass a route determines the clarity of their footprints. So if there are only a few ants on that route, then the density of ants on that route is reduced and there will be no ants at all. Conversely, if the route taken, the number of ants is large, the density of ants that pass through it will increase, or even all ants will go through the trajectory. While [5] in his research to predict travel time on the network in 3 cities with ACO, using 1000 ants, because the number of nodes on the network of 3 cities is a lot.

While in this study, the agent used was 100 ants, with 50 iterations, with consideration [5], the number of nodes was only 27 and the number of possible alternative routes defined was 9. While the use of the number of iterations of 50 was done so that the alternative routes produced were more varied. and consideration of running time. Then filter the alternative paths generated from those 100 agents, so that there are no repeated paths. Filtering is done when the system is looking for travel time, for all paths traversed by each agent. The system will filter the travel time so that there is no repetition of time, and the travel time of the selected alternative route must be smaller than the travel time of the main route, which is 71 minutes. The implementation of ACO is carried out in several scenarios of alpha and beta values. The selected alternative paths issued by the ACO algorithm are shown in Table 4.

Table 4 shows the number of alternative paths generated by the ACO algorithm in one run with several scenarios for alpha and beta values. If seen from the table, ACO with beta value = 2 produces more alternative paths than ACO with beta value = 8. So it can be said that the smaller the beta value, the more various alternative pathways will be generated.

Table 4. Number of alternative route in each scenario

Scenario		Alternative Route
Alpha	Beta	
0.15	2	d, e, g, h, i
0.15	8	d, e, h
0.5	2	d, e, g, h, i
0.5	8	d, e, g
0.85	2	d, e, g, h, i, j
0.85	8	d, e

In the implementation of the Firefly Algorithm, the number of firefly used is as much as the number of existing nodes, namely 28 firefly. A series of nodes formed by FA randomly is called a population. This population will continue to change, until the nodes in the population from the best firefly. That is, in the population there are nodes that form a series of alternative paths.

If the ACO, the system will filter the travel time to get an alternative path, then in the implementation of the FA algorithm, the population filtering system formed based on the last firefly light intensity is greater than zero, so that the alternative path obtained is not the main route. So, when the population has not formed an alternative path and the light intensity value is still zero, a change in the position of the firefly will be made.

The firefly position change is done in 5 iterations. The first iteration is to find the position of the firefly, which forms one of the alternative paths. When the next node is a neighbor to the current firefly position, where the current firefly position with the next node has a distance or edge, it will be included in the best Firefly array. But if it is not a neighbor, it will enter the second iteration, which is to check until firefly currently encounters a node that is a neighbor. When this has happened, it will enter the third iteration, which is to repeat the same as the first iteration. When a firefly has found a node that is not a neighbor, it will enter the fourth and fifth iterations. This loop does the same thing as the previous loop, that is, if the next node is a neighbor, then that node will be included in the best Firefly and search again for the next node, to the last node in the population. If they are not neighbors, then these nodes will be entered directly into the best Firefly and will continue to the next iteration. An explanation of the firefly movement is presented in Appendix D.

While Table 5 shows the number of alternative paths produced by FA with 3 scenarios, for each alpha value produces 3 routes. FA produces a different path for each alpha value, although there is 1 route produced by all scenarios.

Table 5. Number of alternative path in each scenario

Alpha	Alternative Route
0.1	b, e, j
0.5	b, c, j
1	c, f, j

3.1.2 Comparison of Alternative Pathway Results between Ant Colony Optimization and Firefly Algorithm

The process of determining alternative routes when the system is running, between ACO and FA there is a difference. System running on ACO is done once for each scenario, while FA, for each scenario is done 3 times. When viewed based on the running time of the system to obtain an alternative route, ACO takes 15 minutes once running, while FA takes 1 minute once running. The ACO's running time. depending on the number of iterations performed. The more iterations, the longer the system will run. Whereas in FA, the running time depends on the number of repetitions of the firefly movement. This is because, prior to the movement of the fireflies, it is necessary to detect the path distance between the nodes to be formed. Table 6 is the alternative path generated by the two algorithms from all scenarios that have been executed.

Table 6. Distance and Travel Time of Alternative Route of ACO and FA

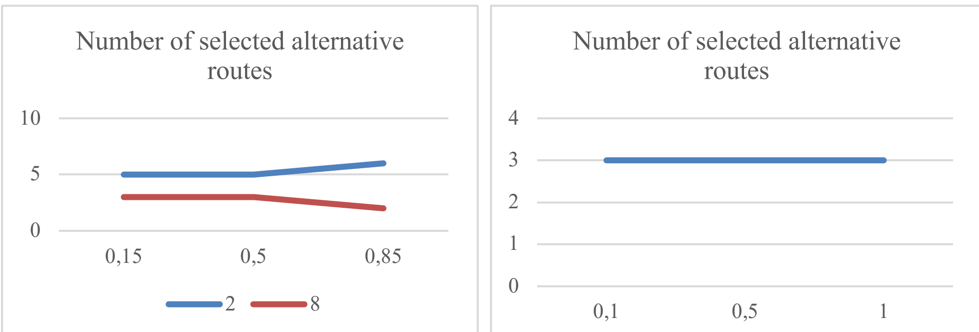
Ant Colony Optimization			Firefly Algorithm		
Alternative Route	**Distance (m)**	**Travel Time (minute)**	**Alternative Route**	**Distance (m)**	**Travel Time (minute)**
E	8720	26	j	10693	32
D	9620	28	e	8720	26
H	8170	24	b	8510	25
G	9070	27	c	7610	22
I	12743	38	f	6530	19
J	10693	32			

The alternative paths in Table 6, generated by the ACO and FA algorithms, are 67% different. The vehicle travel time on the selected alternative route with ACO and FA, is shorter than the vehicle travel time on the main route, which is 71 minutes. Based on table 6, it can be concluded that the alternative route that has the most optimal travel time and distance from ACO is the alternative route h (Sukapura-Batununggal-Cijagra route) with a travel time of 24 minutes and a distance of 8170 meters. As for FA, the optimal alternative route is line f (Telkom-Sukapura-Yogya-Buah Batu Line), which takes 19 minutes and a distance of 6530 meters.

3.2. Discussion

3.2.1 Parameter Value Effect Test

Figure 4 presents the effect of using different parameters on the number of paths generated by the two algorithms. In Figure 3(a), it can be seen that in ACO, the alpha parameter has no effect on the number of alternative paths produced, but the beta parameter is very influential. The greater the beta value, the fewer the number of paths generated. While in Figure 3(b), it shows that in FA, the alpha parameter has no effect on the number of alternative paths produced.



(a) ACO (b) FA

Figure 3. The effect of parameter values on the number of selected routes

Likewise, when tested using analysis of variance, which is a test to determine the significance of the effect of variation scenarios from alpha and beta parameters on ACO and FA. The use of analysis of variance (ANOVA) in this study, is based on research [26], where to determine the effect of several scenarios used in the implementation of ACO, two-factor analysis of variance (ANOVA) was used. ANOVA was performed on each scenario of the three problems, to determine whether there were differences in route distance as a result of using different and single versus multiple colony approaches. The results obtained from ANOVA are significant differences in the size of the candidate list and the optimization method used, the size of the candidate list is significantly different in finding route distances. While in this study the use of ANOVA was carried out to determine the effect of using alpha and beta parameters on ACO and the firefly algorithm, on determining alternative routes based on distance and time. Table 8 is the result of the analysis of variance, which shows the effect of the parameters on the alternative routes generated by ACO. While Table 7 is the result of the analysis of the variance of the influence of the parameters on the alternative routes produced by FA

Table 7. ANOVA of ACO Parameter Scenarios

Source of variation	Sum Square	Degree of free	Mean Square	F Value	P Value	F Critis
Alpha	0.6293	2	0.3146	2.5002	0.2855	19
Beta	6.7557	1	6.7557	53.7275	0.0181	18.5128
Error	0.2515	2	0.1257			
Total	7.6365	5				

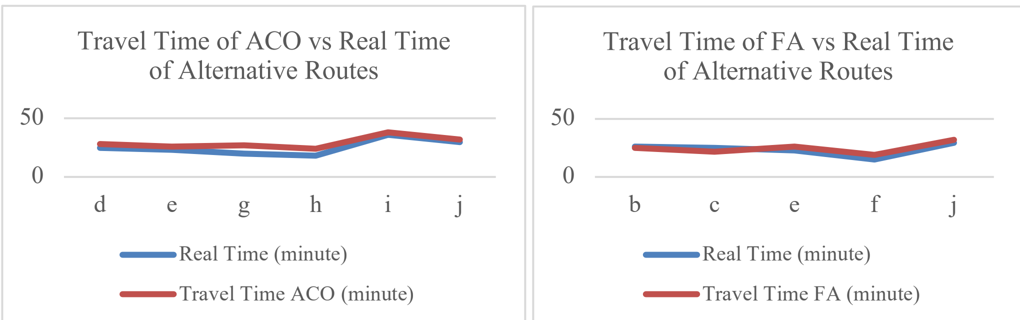
Table 7 shows that only the beta parameter has a significant effect on the number of alternative routes generated by ACO, with a 95% confidence interval. But the variation of the alpha parameter has no effect on the number of selected routes. Likewise, in Table 8 it can be seen that the alpha parameter has no effect on the number of alternative routes generated by the FA algorithm.

Table 8. ANOVA of FA Parameter Scenarios

Source of variation	Sum Square	Degree of free	Mean Square	F Value	P Value	F Critis
Alpha (between groups)	16.8889	2	8.4444	0.2912	0.7574	5.1433
Within groups	174	6	29			
Total	190.8889	8				

3.2.2 Accuracy of alternative routes selection with ACO and FA

Figure 4(a) is a graph that shows the comparison between the travel time generated by the ACO algorithm and the real time on the direct road. While Figure 4(b) is a comparison graph between the travel time generated by the FA algorithm with real time on the direct road. From the two figures, it can be explained that the travel time produced by the FA algorithm is closer to real time on the road.



(b) ACO (b) FA

Figure 4. Travel Time vs Real Time of Alternative Routes

The statement above is also supported through accuracy testing based on the RMSE value. The accuracy check of the process of determining the selected alternative paths produced by ACO and FA is carried out by comparing the travel time generated by each algorithm with the travel time in the field, using RMSE. The RMSE value of the travel time and real time of the alternative route generated by ACO is 4.65, while the RMSE value of the FA algorithm is 2.64. This explains that FA is more accurate in selecting an alternative route as a solution when the main line is experiencing congestion. This statement is also supported by the large percentage accuracy of the FA of 88%, this value is greater than the accuracy of ACO.

3.2.3 Performance comparison of ACO and FA algorithm

Based on the results discussed above, Table 9 presents a performance comparison between ACO and the FA algorithm, in the process of determining alternative routes for cars as a solution to traffic congestion. The performance comparison includes the best alternative route, alternative route search process, the average running time, the number of alternative routes selected, and the effect of parameter variation scenarios.

Table 9. ACO and FA Performance related Parameter Scenario

No	Performance	ACO	FA
1	The best alternative route	The alternative route h (Sukapura-Batununggal-Cijagra route) with a travel time of 24 minutes and a distance of 8170 meters	The optimal alternative route is line f (Telkom-Sukapura-Yogya-Buah Batu Line), which takes 19 minutes and a distance of 6530 meters
2	Alternative route search process	The travel time is filtered, where the travel time that is smaller than the main route's travel time will be an alternative route	Filtering is carried out at the last firefly light intensity value must be greater than zero
3	The average running time	(15 minute), depends on the number of iterations, meaning that the more iterations, the longer the system running time	(1 minute), depends on movement firefly, namely the system running time can be fast or even long
4	The number of alternative routes selected	Can provide alternative route for more than one lane in 1 run	Only provides one lane
5	The effect of parameter variation scenarios	The lower the beta value, the more various alternative routes ACO can generate in all scenarios	Can produce the same alternative route for all scenarios

In the case study in this research, in general the FA algorithm is superior in determining alternative routes, with consideration of alternative routes resulting in shorter distances and shorter travel times, shorter running times, no influence on alpha parameter values. But in each iteration, the number of alternative routes generated is less.

4 CONCLUSION

Based on the results of testing and parameter optimization analysis carried out on the development of an alternative route search system, using Ant Colony Optimization and Firefly Algorithm in the Bojongsoang to Buah Batu (BKR) area and based on Figure 4, it was concluded that based on the travel time in determining alternative routes produced by FA was superior to ACO, with an accuracy of 88%. This is also supported by the performance of the FA algorithm which is generally superior, where the resulting alternative route is shorter in distance and shorter travel time, running time is shorter, there is no influence on the alpha parameter value. But in each iteration, the number of alternative routes generated is less. The best alternative route produced by ACO is the alternative route h (Sukapura-Batununggal-Cijagra route) with a travel time of 24 minutes, while FA produces the best alternative route, namely route f (Telkom-Sukapura-Yogya-Buah Batu route) with a travel time of 19 minutes.

5 REFERENCES

[1] A. Marsiela, "ADB: Kemacetan di Bandung Melebihi Jakarta." https://www.beritasatu.com/nasional/578854-adb-kemacetan-di-bandung-melebihi -jakarta (accessed Jun. 23, 2021).
[2] R. Refianti and A. Benny, "Solusi optimal travelling salesman problem dengan Ant Colony System (ACS)," no. February 2016, 2005, doi: 10.13140/RG.2.1.2089.7047.
[3] Y. Siyamtining Tyas and W. Prijodiprodjo, "Aplikasi Pencarian Rute Terbaik dengan Metode Ant Colony Optimazation (ACO)," *IJCCS (Indonesian J. Comput. Cybern. Syst.*, vol. 7, no. 1, p. 55, 2013, doi: 10.22146/ijccs.3052.
[4] V. Danchuk, O. Bakulich, and V. Svatko, "An Improvement in ant Algorithm Method for Optimizing a Transport Route with Regard to Traffic Flow," *Procedia Eng.*, vol. 187, pp. 425–434, 2017, doi: 10.1016/j.proeng.2017.04.396.
[5] R. Claes and T. Holvoet, "Ant Colony Optimization applied to route planning using link travel time predictions," *IEEE Int. Symp. Parallel Distrib. Process. Work. Phd Forum*, no. April 2014, pp. 358–365, 2011, doi: 10.1109/IPDPS.2011.173.
[6] A. Bolufé-Röhler, J. M. O. Pereira, and S. Fiol-González, "Traffic flow estimation using ant colony optimization algorithms," *Comput. y Sist.*, vol. 18, no. 1, pp. 37–50, 2014, doi: 10.13053/CyS-17-4-2013-017.
[7] L. Groleaz *et al.*, "ACO with automatic parameter selection for a scheduling problem with a group cumulative constraint To cite this version : HAL Id : hal-02531062 ACO with automatic parameter selection for a scheduling problem with a group cumulative constraint," 2020.

[8] H. Fahmi, M. Zarlis, E. B. Nababan, and P. Sihombing, “Ant Colony Optimization (ACO) Algorithm for Determining the Nearest Route Search in Distribution of Light Food Production,” *J. Phys. Conf. Ser.*, vol. 1566, no. 1, 2020, doi: 10.1088/1742-6596/1566/1/012045.
[9] L. Groleaz, S. N. Ndiaye, and C. Solnon, “ACO with automatic parameter selection for a scheduling problem with a group cumulative constraint,” *GECCO 2020 - Proc. 2020 Genet. Evol. Comput. Conf.*, pp. 13–21, 2020, doi: 10.1145/3377930.3389818.
[10] P. N. Ky Phuc and N. Le Phuong Thao, “Ant Colony Optimization for Multiple Pickup and Multiple Delivery Vehicle Routing Problem with Time Window and Heterogeneous Fleets,” *Logistics*, vol. 5, no. 2, p. 28, 2021, doi: 10.3390/logistics5020028.
[11] N. Ali, M. A. Othman, M. N. Husain, and M. H. Misran, “A review of firefly algorithm,” *ARPN J. Eng. Appl. Sci.*, vol. 9, no. 10, pp. 1732–1736, 2014.
[12] M. A. Tawhid and A. F. Ali, “Direct search firefly algorithm for solving global optimization problems,” *Appl. Math. Inf. Sci.*, vol. 10, no. 3, pp. 841–860, 2016, doi: 10.18576/amis/100304.
[13] X. Wang, M. He, and H. Jiang, “A Discrete Firefly Algorithm for Routing Optimization of Milk-Run,” no. Icadme, pp. 1538–1543, 2015, doi: 10.2991/icadme-15.2015.285.
[14] J. Sudirwan, S. N. Fadlilah, and T. Teguh, “Aplikasi Hybrid Firefly Algorithm untuk Pemecahan Masalah Traveling Salesman: Studi Kasus pada PT Anugerah Mandiri Success,” *ComTech Comput. Math. Eng. Appl.*, vol. 5, no. 2, p. 828, 2014, doi: 10.21512/comtech.v5i2.2281.
[15] R. Micale, G. Marannano, A. Giallanza, P. P. Miglietta, G. P. Agnusdei, and G. La Scalia, “Sustainable vehicle routing based on firefly algorithm and TOPSIS methodology,” *Sustain. Futur.*, vol. 1, no. September, p. 100001, 2019, doi: 10.1016/j.sftr.2019.100001.
[16] J. Kwiecień and B. Filipowicz, “Firefly algorithm in optimization of queueing systems,” *Bull. Polish Acad. Sci. Tech. Sci.*, vol. 60, no. 2, pp. 363–368, 2012, doi: 10.2478/v10175-012-0049-y.
[17] A. Khadwilard, S. Chansombat, T. Thepphakorn, and P. Thapatsuwan, “Application of Firefly Algorithm and Its Parameter Setting for Job Shop Scheduling,” no. January, 2012.
[18] M. M. Gangadharan and A. Salgaonkar, “Ant colony optimization and firefly algorithms for robotic motion planning in dynamic environments,” *Eng. Reports*, vol. 2, no. 3, pp. 1–23, 2020, doi: 10.1002/eng2.12132.
[19] W. Windarto and E. Eridani, “Comparison of particle swarm optimization and firefly algorithm in parameter estimation of lotka-volterra,” *AIP Conf. Proc.*, vol. 2268, no. September, 2020, doi: 10.1063/5.0017245.
[20] D. Kurniawan and A. A. A. Colony, “93603-ID-none,” vol. 4, no. 3, 2016.
[21] S. Sawyer, “Analysis of Variance : The Fundamental Concepts,” no. December, 2017, doi: 10.1179/jmt.2009.17.2.27E.
[22] W. Wang and Y. Lu, “Analysis of the Mean Absolute Error (MAE) and the Root Mean Square Error (RMSE) in Assessing Rounding Model,” *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 324, no. 1, 2018, doi: 10.1088/1757-899X/324/1/012049.
[23] A. J. Umbarkar, U. T. Balande, and P. D. Seth, “Performance evaluation of firefly algorithm with variation in sorting for non-linear benchmark problems,” *AIP Conf. Proc.*, vol. 1836, no. June 2017, 2017, doi: 10.1063/1.4981972.
[24] Lestari Himmawati Puji dan Eminugroho Ratna Sari, “Penerapan algoritma koloni semut untuk optimisasi rute distribusi pengangkutan sampah di kota Yogyakarta,” *J. Sains Dasar*, vol. 2, no. 1, pp. 13–19, 2014, doi: 10.21831/jsd.v2i1.2373.
[25] S. Katiyar, Ibraheem, and A. Q. Ansari, “Ant Colony Optimization : A Tutorial Review Ant Colony Optimization : A Tutorial Review Department of Electrical Engineering Corresponding Author : (Email : aqansari@ieee.org),” no. August, 2015.
[26] J. E. Bell and P. R. McMullen, “Ant colony optimization techniques for the vehicle routing problem,” *Adv. Eng. Informatics*, vol. 18, no. 1, pp. 41–48, 2004, doi: 10.1016/j.aei.2004.07.001.

Appendix A Multiple Alternative Routes of Route Modeling Results

ID	Route
a	1-2-3-4-5-6-7-8-9
b	1-2-3-10-11-12-13-14-15-7-8-9
c	1-2-3-10-11-12-13-14-15-8-9
d	1-2-3-10-11-16-5-6-17-18-14-15-7-8-9
e	1-2-3-10-11-16-5-6-17-18-14-15-8-9
f	1-2-3-10-11-16-5-6-7-8-9
g	1-2-3-4-5-6-17-18-14-15-7-8-9
h	1-2-3-4-5-6-17-18-14-15-8-9
i	1-2-19-20-21-22-23-34-25-26-7-8-9
j	1-2-3-4-27-21-22-23-24-25-26-7-8-9

Appendix B Stages of ACO Implementation

NO	Activity	Information
1	Initialize ACO parameters	a. Distance between nodes b. Iteration = 50 c. Agent = 100 d. Node = 27 e. Alpha = 0.5 f. Beta = 2 g. Evaporation = Random Value h. Speed (alternative route) = 333.33 (20 km/hr) i. Speed (main route) = 83.333 (5 km/hr) j. Initial pheromone = 0.01
2	Calculating visibility	If the distance between nodes is 0, then the visibility is also 0
3	Perform agent transition	When nodes I and j are connected by an edge, the agents at node I can move to node j. Transition stops when it reaches the destination node
4	Calculation of distance and time	If all agents have arrived at the destination node
5	Time selection	The time obtained from each agent will be taken the shortest time as an alternative route
6	Update the Pheromone value	Update is done using the (3) equation where the change in pheromone
7	Repeating	Steps 3 - 6 is repeated 50 times
8	Converting an array of alternative routes to a list {1, 2, 3, 10, 11, 16, 5, 6, 17, 18, 14}	Issue several alternative routes until the destination node with the fastest travel time is obtained

Appendix C Stages of Firefly Algorithm Implementation

NO	Activity	Information
1	Initialize FA parameters	a. Distance between nodes b. n population = 7 c. n Gen = 27 d. lb = 4 e. ub = 27 f. Alpha = 1 g. Theta = 0.97 h. Speed (alternative route) = 333.33 (20 km/hr) i. Speed (main route) = 83.333 (5 km/hr)
2	Initialization of firefly population	Randomly based on the number of random permutation n populations
3	Calculation of light intensity	Using the (4) equation
4	Check the light intensity for each last firefly is more than 0	If the intensity is more than 0, then firefly is a solution, if not, then iterate again until a solution is obtained
5	Switching firefly	Checking between the current firefly and the next firefly there is a path or not. If so, the current firefly will move to the next firefly. Otherwise, firefly will be moved according to the SwapWithRouteDetection function
6	Repetition of points 4 and 5	Until a firefly sequence is found in a population is an alternative path

7	Best firefly array representation and alternative routes	Best firefly {1, 2, 3, 10, 11, 12, 13, 14, 15, 8, 9, 6, 17, 26, 25, 6, 21, 19, 27, 24, 20, 18, 4, 16, 22, 23, 7} Alternative routes {1, 2, 3, 10, 11, 12, 13, 14, 15, 8, 9}

Appendix D The Firefly Movement

Population	[1, 2, 3, 4, 25, 17, 26, 5, 15, 19, 6, 14, 27, 9, 20, 21, 8, 16, 22, 13, 18, 23, 10, 12, 24, 11, 7]
Best Firefly	[1]
1st iteration (i index starting from 0)	1. The 1st index node in the population is 2, 1 to 2 has a distance, then 2 enters [1, 2] 2. The 2nd index node in the population is 3, 2 to 3 has a distance, then 3 enters [1, 2, 3] 3. The 3rd index node in the population is 4, 3 to 4 has a distance, then 4 is in [1, 2, 3, 4] 4. The 4th index node in the population is 25, 4 to 25 has no distance, then the 1st loop stops
Best Firefly	[1, 2, 3, 4]
2nd iteration (j index starting from i+1)	1. The 4th index node in the population is 25, 4 to 25 has no distance, then proceed to check the next node 2. The 5th index node in the population is 17, 4 17 to no distance then continue checking to the next node 3. The 6th index node in the population is 26, 4 to 26 has no distance then continue checking to the next node 4. The 7th index node in the population is 5, 4 to 5 has a distance, so the 2nd loop stops
Best Firefly	[1, 2, 3, 4]
3rd iteration (k-th index starts from j+1)	1. The 7th index node in the population is 5, 4 to 5 has a distance, then 5 is in [1, 2, 3, 4, 5] 2. The 8th index node in the population is 15, 4 to 15 has no distance, so the 3rd iteration stops
Best Firefly	[1, 2, 3, 4, 5]
4th iteration (lth index starting from i+1 to j)	1. The 4th index node in the population is 25, 4 to 25 has no distance, then continue checking to the next node 2. The 5th index node in the population is 17, 4 17 to no distance, then continue checking to the next node 3. The 6th index node in the population is 26, 4 to 26 has no distance, then continue checking to the next node node
Best Firefly	[1, 2, 3, 4, 5, 25, 17, 26]
5th iteration (m-th index starts from k to finish)	[1, 2, 3, 4, 5, 25, 17, 26, 15, 19, 6, 14, 27, 9, 20, 21, 8, 16, 22, 13, 18, 23, 10, 12, 24, 11, 7]