ISSN 2278-3091
Volume 8, No.5, September - October 2019
International Journal of Advanced Trends in Computer Science and Engineering
Available Online at http://www.warse.org/IJATCSE/static/pdf/file/ijatcse121852019.pdf
https://doi.org/10.30534/ijatcse/2019/121852019

Traffic Route Optimization for University Students with Enhanced Ant Colony Optimization Algorithm

Raja Ameerul Eizamuddin Raja Aisham[1], Marina Yusoff[2], Norhidayah A Kadir[3]
[1]Faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA, 40450 Shah Alam, Selangor, MALAYSIA, rajameerol@gmail.com
[2]Advanced Analytic Engineering Center (AAEC), Faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA, 40450 Shah Alam, Selangor, MALAYSIA, marinay@tmsk.uitm.edu.my
[3] Faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA, 40450 Shah Alam, Selangor, MALAYSIA, norhidayah@tmsk.uitm.edu.my

ABSTRACT

Ant Colony Optimization (ACO) and its variants that mimic an ant traveling behavioral still obtain good employment demand for various real-world problems. This paper highlights a new improvement of ACO algorithms with respects to the ant searching experience considering the possible route from the current location to the destination and obstacles that can be avoided to reduce travel delays. Datasets of students travelling from their residences to the university were used. The findings demonstrate the performance of the enhanced ACO algorithm with a difference ACO parameter to determine the shortest travelling time to reach the destination based on occurrences of delay time as such of traffic light, roundabout and intersections.

Key words: Ant Colony Optimization Algorithm, ACO, Google Map API, Optimal Route Navigation.

1. INTRODUCTION

Many solutions to improve traffic routing have been using biologically inspired techniques [1]–[3] such as particle swarm optimization (PSO), ant colony optimization (ACO), elephant search, genetic algorithm (GA) - to name a few. Each technique has their own strength, limitation and suitability for each of the problem behavior. Increasing in popularity, ACO is a metaheuristic method based on nature-inspired algorithm and a decentralized system for problem solving characterized by self-organization, flexibility and robustness [4]. Since ACO is based on ant behavior, the path taken is chosen randomly because the ant has no clue for the best path [5]. In larger graphs, ACO is expected to perform faster than Dijkstra's algorithm for the reason of continuous increasing/changing edges and nodes in the process of finding the optimal path [6]. Rising traffic congestion stemming from the increasing number of road users makes traffic routing a crucial everyday task. Although, lots of effort has been put in finding optimal routes, yet improvements are still required.

Recently, there are new navigation technologies with real-time traffic information, which provides information about real time alternate routes in assisting drivers to select optimal routes [7]. Based on a survey done, these navigation systems such as Google Maps and Waze are reported widely using on smartphone [8].

The Google Maps application is using Dijkstra's Algorithm to find the shortest path and make route-finding possible [9]. It provides variety of transportation options as well as additional information such as businesses and monuments. Waze application on the other hand, is using Routing Algorithm with a combination of machine learning and human refinement [10]. Focusing on social network for information, drivers can share information about accidents, roadblocks, road closures, and many more. It is solely for navigation by car. Both Google Maps and Waze are belonging to Google [11].

This paper highlights the use of an improved ACO, focusing on datasets of routing from students' residences to the Faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA, Malaysia. Implementation of ACO algorithm as an alternative solution to minimize delay time for the university students attending classes is proposed. The identification of an optimal route and the best time to depart are of prime important to ensure they are on time. The use of the ACO algorithm that can solve optimization problem is expected to reduce or minimize the delays of student coming late to class, considering the possible route from their current location and obstacle that can be avoided to reduce the delays.

1.1 Factors that Effects the Travelling Time

Factors concerning travelling time are a number of vehicles, number of lanes, traffic lights, roundabout and intersection [12]. In this study, numbers of vehicles on the road were excluded to eliminate additional factors causing traffic delays, ergo; factors considered are traffic light, roundabout and intersection.

Time delay varies according to the factors mentioned above. Moving vehicle might need to slow down, stop for a while or absolutely stop moving. Time delay for each road junction were obtained based on research [13-15] related with the road junction and is based on direct observation within area Seksyen 7. Each road junction delay that were collected from the related articles and observation is shown in Table 1.

Table 1: Time delay collected from article and observation

Road junction	Time delay from articles, [12-14]	Time delay from observations
Traffic light	42	10
Roundabout	12.1	4
Intersection	7.1	1.5

2. AN IMPROVED ACO ALGORITHM

The ACO parameters to be initialized are the number of iterations, number of ants, evaporation rate, alpha rate, beta rate and other fundamental ACO parameter. Equation (1) is the probability formula consisting of heuristic information and pheromone information that will be used for the ant to move from their current point to the next point. The formula helps in finding possible point in the movement of ants. The forward ants will select a random node for the first time taking the information from the distance matrix. Ants who that successfully reached the destination point will update the pheromone value at the edges.

$$p_{i,j} = \frac{\left(\tau_{i,j}^{\alpha}\right)\left(\frac{1}{w_{i,j}^{\beta}}\right)}{\sum\left(\tau_{i,j}^{\alpha}\right)\left(\frac{1}{w_{i,j}^{\beta}}\right)} \qquad (1)$$

$p_{i,j}$ stands for probability formula between point *i* to point *j*. $\tau_{i,j}$ in the formula above is the amount of pheromone on edge between point *i* to point *j*, while the w_{ij} is the weight or length between point *i* to point *j*. α is a parameter to control the influence of pheromone and β is a parameter to control the influence of the edge weight.

Equation (2) is the mathematical equation to update each amount of pheromone after each ant has find the possible path to reach the destination. It is as a progressive strategy for updating the pheromone trails. The progressive strategies can be defined as a pheromone trail that were reinforced with each route that the ant has run through from one point to another point during the ant's path searching to reach the destination point.

$$\tau_{i,j}^{k} = \tau_{i,j}^{k} + \Delta\tau_{i,j}^{k} \qquad (2)$$

where

$$\Delta\tau_{i,j}^{k} = \begin{cases} \frac{1}{L_k} & \text{if ant k travel on point i,j} \\ 0 & \text{otherwise} \end{cases}$$

$\tau_{i,j}$ can be defined as the pheromone trail value for point *i* to point *j*. The previous pheromone will be added with the latest growth of the pheromone trail, $\Delta\tau_{i,j}^{k}$. The latest growth, $\Delta\tau_{i,j}^{k}$ of the pheromone trail can be computed by dividing 1 with L_k. However, if the ant does not travel between point *i* to point *j*, the growth of pheromone is 0. Pheromone will be updated whenever each ant has successfully reached the destination point.

Equation (3) is the equation to update the pheromone matrix with the evaporation rate. High evaporation rates result in exploration behaviors of the ants, which results in ants getting lost. Conversely, low evaporation rates result in exploitation behaviors, which would lead to in an inability to acquire the optimal path. Thus, the adaptation capabilities of ACO rely on the pheromone evaporation mechanism. The rate used is usually fixed. Pheromone trails which representing the bad solutions from previous environment can be eliminated by the pheromone evaporation [16]. The pheromone evaporation rate is utilized using following equation:

$$\tau_{i,j}^{k} \leftarrow (1-\rho)\,\tau_{i,j}^{k} \qquad (3)$$

The above equation will update the pheromone trail value by calculating the evaporation rate of the pheromone. ρ is the constant value of the evaporation rate that will be minus by 1. The result will be multiplied to the current pheromone value. Each evaporation rate will be update before each iteration ends.

Based on ACO algorithm above, objective function for shortest travelling time based on the duration of the travelling time and road junction delay time is formulated as in (4).

$$\text{obj} \quad F = \sum_{i=1} D_{P_{i(\text{best})}} + DT \qquad (4)$$

where

$$DT = \left(\sum \text{TLight} \times \text{TLightDT}\right) + \left(\sum \text{RBout} \times \text{RBoutDT}\right) + \left(\sum \text{Inter} \times \text{InterDT}\right) \qquad (5)$$

where

DT	:	Total delay time
TLight	:	Number of traffic light on the path
TLightDT	:	Traffic light delay time
RBout	:	Number of roundabouts on the path
RBoutDT	:	Roundabout delay time
Inter	:	Number of intersections on the path
InterDT	:	Intersection delay time

The initial procedure for initialization of the distance matrix, pheromone matrix as well as the formula is based from article [17]. Thus, with few adjustments made, an improved ACO is shown below.

```
   begin
1      Initialize parameter;
2      Initialize distance matrix;
3      Initialize pheromone matrix;
4      while number of iterations < maximum number of
       iteration do
5          for number of ants < maximum number of ants do
6              repeat
7                  Position each ant in a starting node;□
8                  Each ant can only run through each node
                   only once;
9                  Choose next node for each ant by applying
                   (1);
10                 Each ant's path is stored in their own
                   memory ;
               until
                   There is no possible next node OR each
                   ant reaches the destination node using the
                   same path;
11             if An ant reach back to it's starting node using
               the same path
               then
12                 Calculate the total traveling time using
                   (4);
13                 Increase the pheromone amount using (2)
                   for the path used;
               end
14             Update pheromone with evaporation rate
               using (3);
15             Analyze the pheromone matrix to identify the
               shortest path based on the objective function;
           end
       end
   end
```

Algorithm 1: An Improved ACO Algorithm

The pseudocode above shows that in each iteration, the number of ants will randomly pick their path until it reaches the destination or dead end (no possible next point). The pheromone will be updated for each node and will then be evaporated at the end of each loop. The improvement can be seen in line 10 and 12. Line 10 considers whether an ant will stop if there is no possible next node or if will stop because it reaches the destination. It will use the same path that was stored in their memory to reach the initial node again. Line 12 on the other hand, considers each ant that reaches back the initial node and (4) is used to calculate the total travelling time.

3. RESULTS AND FINDINGS

3.1 Datasets

Possible routes in Seksyen 7, Shah Alam to reach Faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA is obtained by using Google Maps API to collect students' residences (staying in Section 7) coordinate. The coordinates are used as the starting point for each path to reach the destination. Further to this, another layer need to be created as a loop condition to start and end the loop. The angle of the square border will be used as a loop condition to make sure that the latitude and longitude is within the red border. The angle of the square border is [{3.091156, 101.475057}, {3.091156, 101.507846}, {3.062571, 101.475057}, {3.062571, 101.507846}].

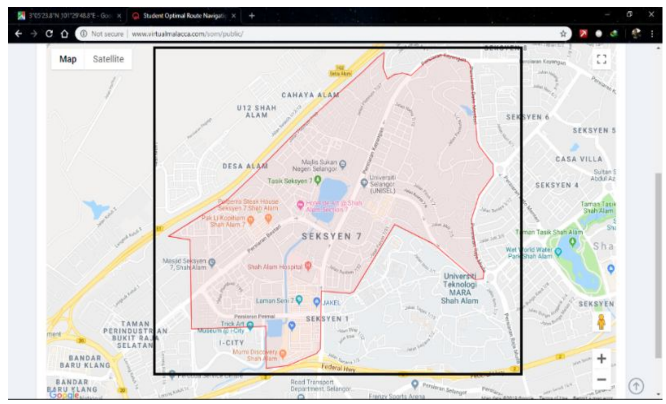

Figure 1: Square border to collect every coordinate that contain in the Seksyen 7 border

The coordinates obtained were then verified using Google Maps Geometry Library and is stored in the form of table. The coordinates are used as the origin location in finding possible routes using Google Maps Navigation. There were 2079 number of different paths and 2468 number of different routes were collected. Five datasets were established and experimented and is demonstrated in the next section.

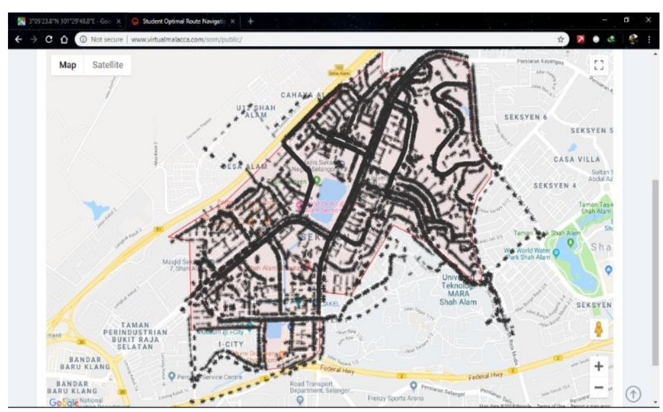

Figure 2: Collection of possible routes based on every coordinate in Seksyen 7

3.2 Parameter Settings

Table 2 listed the parameter settings for the computational experiments. Datasets were taken from data discussed in subsection 3.1. The computational results are based on various numbers of artificial ants, number of iterations, alpha rate, beta rate, evaporation rate and time delay parameter to evaluate performances of the algorithm. The objective function aims for a minimum travelling time.

Table 2: Parameter settings

Parameter		Values
Number of ants		10, 30, 50, 100
Number of iterations		10, 20, 30, 40
Alpha rate		0.3, 0.7, 1, 2
Beta rate		0.3, 0.7, 1, 2
Evaporation rate		0.01, 0.3, 0.7, 0.9
Time delay	Traffic light	0, 10, 42
	Roundabout	0, 4, 12.1
	Intersection	0, 1.5, 7.1

The computational experiment result is shown in Table 4. The travelling time and computational time is elaborated in the following section.

3.3 Computational Results based on Travelling Time

Constant value used for ACO parameters are as follow :

Table 3: Fixed ACO parameter on difference performance testing

Number of iterations	Alpha rate	Beta rate	Evaporation rate
20	0.7	0.3	0.01

The stopping condition is based on either all vehicles arrived at destination or when its reach 200 iterations.

Based on the proposed ACO algorithm, travelling time was measured in second (see Table 4). The results are categorized by the ACO parameter and time delay. Travelling time as calculated using (4), were the main objective in this research.

The result demonstrates that some of the ACO parameter and time delay parameter can affect the result of the travelling time. The ACO parameter has its affect on the optimal path that needs to be taken. Time delay parameter on the other hand, is seen to be affecting the result of the travelling time. The larger the path size provided, the more path selection and possible route. The time delay parameter will have a slight change on the result of travelling time.

However, the performance issue is a crucial part in processing optimization algorithm because it can affect the user experience to process the optimization algorithm completely. Thus, if the issue is not resolve quickly, it can slow the device or network.

4. CONCLUSION

To recapitulate this research, the aim of this research is to find the shortest travelling time between the student residence and the FSKM building at UiTM Campus Shah Alam by utilizing the data collected using Google Maps API to test and validate the optimization algorithm. Several parameter tunings and testing were made to evaluate the performance of ACO using different datasets of possible routes. In the nutshell, the result of the ACO performance evaluation can determine the optimal ACO parameter value in reaching the optimal performance.

Table 4: Travelling time computational result

Dataset	Path Size	Number of artificial ants				Number of iterations				Alpha rate				Beta rate				Evaporation rate			
		10	30	50	100	10	20	30	40	0.3	0.7	1	2	0.3	0.7	1	2	0.01	0.3	0.7	0.99
A	25	575	575	575	579	650	575	575	579	559	575	559	567	575	575	575	575	575	567	624	623
	50	578	667	593	609	605	578	605	685	630	578	685	672	578	605	609	643	578	605	578	728
	75	666	575	559	575	579	666	530	583	563	666	649	563	666	728	650	725	666	712	530	575
	100	654	563	559	570	587	654	536	554	536	654	570	536	654	623	639	634	654	536	652	769
B	25	620	735	858	738	620	620	744	858	620	620	815	613	620	735	821	782	620	782	782	670
	50	677	735	738	738	662	677	771	738	738	677	738	707	677	735	772	788	677	708	636	757
	75	662	735	735	738	561	662	760	760	880	662	846	760	662	672	764	904	662	561	677	772
	100	712	642	735	738	744	712	768	626	833	712	702	621	712	797	748	-	712	790	833	854
C	25	455	455	455	455	455	455	455	455	455	455	455	424	455	449	455	455	455	424	424	509
	50	455	419	455	455	455	455	455	455	455	455	419	501	455	455	540	455	455	455	540	455
	75	455	455	455	455	455	455	455	455	455	455	455	455	455	455	410	455	455	455	410	455
	100	455	455	455	455	455	455	455	455	455	455	455	401	455	401	401	-	455	455	455	455
D	25	681	681	681	681	681	681	681	681	681	681	681	660	681	681	681	681	681	681	681	681
	50	721	742	742	742	742	721	742	742	649	721	742	721	721	721	742	742	721	681	742	742
	75	917	903	903	903	917	917	917	903	903	917	810	903	917	885	824	903	917	917	921	885
	100	715	756	663	736	736	715	742	736	756	715	660	704	715	736	758	756	715	681	742	735
E	25	509	440	378	378	497	509	503	378	378	509	503	659	509	378	378	633	509	378	378	535
	50	420	378	378	544	420	420	378	378	378	420	545	506	420	544	378	591	420	378	420	509
	75	378	400	400	378	400	378	420	400	378	378	378	550	378	378	378	378	378	378	400	570

ACKNOWLEDGEMENT

This work was supported by Faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA Shah Alam, Selangor, Malaysia.

REFERENCES

1. S. Kaur, T. C. Aseri, S. Rani. **Qos-aware routing in vehicular ad hoc networks using ant colony optimization and bee colony optimization**, in *Proc. of 2nd International Conference on Communication, Computing and Networking*, Singapore, 2019, pp. 251–260. https://doi.org/10.1007/978-981-13-1217-5_25
2. S. Srivastava, S. K. Sahana. **A survey on traffic optimization problem using biologically inspired techniques**, *Natural Computing*, pp. 1–15, 2019.
3. X. Zhou, **Research on the optimization of tourism traffic routes based on ant colony algorithm,** vol. 32, no. 2, pp. 819–827, 2017.
4. A. Lazarowska. **Ant colony optimization based navigational decision support system,** in *Proc. computer science*, 2014, pp. 1013–1022. https://doi.org/10.1016/j.procs.2014.08.187
5. M. Dorigo, L. M. Gambardella. **Ant colony system: a cooperative learning approach to the traveling salesman problem**, *IEEE Transactions on evolu-tionary computation*, vol. 1, no. 1, pp. 53–66, 1997.
6. H. B. Steinsholt, D. D. Aasen. **Examining ant colony optimiza- tion performance for ship evacuation**, M. S. thesis, University of Agder, 2013.
7. V. Kumar. **Optimum route selection for vehicle navigation**, *International Journal of Advanced Computer Science and Applications*, vol. 7, no. 2, pp. 142-148, 2016. https://doi.org/10.14569/IJACSA.2016.070219
8. V. Rotondi, L. Stanca, M. Tomasuolo. **Connecting alone: Smartphone use, quality of social interactions and well-being**, *Journal of Economic Psychology*, vol. 63, pp. 17–26, 2017.
9. M. Byrn. **The simple, elegant algorithm that makes google maps possible,** *Motherboard*, March 2015.
10. N. Bardin. **Keeping cities moving – how waze works**.
11. P. Cohan. **Four reasons google bought waze**, *Forbes. Com*, June 2013.
12. N. Kuchuk, A. S. Mohammed, A. Shyshatskyi, O. Nalapko. **The Method of Improving the Efficiency of Routes Selection in Networks of Connection with the Possibility of Self-Organization**, *International Journal of Advanced Trends in Computer Science and Engineering*, vol. 8, no. 1.2, pp. 1-6, 2019.
13. T. I. Atomode. **Assessment of traffic delay problems and characteristics at urban road intersections: a case study of Ilorin, Nigeria**, *IOSR Journal of Humanities and Social Science*, vol. 12, no. 4, pp 06-16, 2013. https://doi.org/10.9790/0837-1240616
14. R. Ak ̧celik. **Capacity and performance analysis of roundabout metering signals**, in *TRB National Roundabout Conference*, Vail, Colorado, USA, 2005, pp. 22–25.
15. G. Asaithambi, C. Anuroop. **Analysis of occupation time of vehicles at urban unsignalized intersections in non-lane-based mixed traffic conditions**, *Journal of Modern Transportation*, vol. 24, no. 4, pp. 304–313, 2016.
16. D. Yuvaraj, M. Sivaram, A. Mohamed Uvaze Ahamed, S. Nageswari. **Nature Inspired Evolutionary Algorithm (ACO) for Efficient Detection of DDoS Attacks on Networks,** *International Journal of Advanced Trends in Computer Science and Engineering*, vol. 8, no. 1.4, pp. 44-50, 2019. https://doi.org/10.30534/ijatcse/2019/0781.42019
17. S. Saeedi, I. Mahdavi. **Using ant colony optimization for shortest path problem,** *Optimization*, August 2007.