M Gmail

Roby Setiadi <robysetiadi.psdie.undip@gmail.com>

Paper Published Online- IJMET- Volume 9, Issue 8, 2018

IAEME Editing Department <editing@iaeme.com>   8 September 2018 16.29
Kepada: robysetiadi.psdie.undip@gmail.com

Dear Sir/ Madam,

Your article published online in International Journal of Mechanical Engineering and Technology (IJMET), Volume 9, Issue 8, August, 2018 and attached the file for your reference. Website www.iaeme.com.

Papers Title:
GRAPH-BASED MODELING AND DIJKSTRA ALGORITHM FOR SEARCHING VEHICLE ROUTES ON HIGHWAYS

Link:

http://www.iaeme.com/MasterAdmin/Journal_uploads/IJMET/VOLUME_9_ISSUE_8/IJMET_09_08_137.pdf

Thanks & Regards
IAEME Publication

---

2 lampiran

📄 **137 Sodik Kirono, Muhammad Irfan Arifianto, Reza Eka Putra, Ahmad Musoleh, Roby Setiadi.pdf**
416K

📄 **IJMET_09_08_137.pdf**
649K

INTERNATIONAL JOURNAL OF MECHANICAL ENGINEERING & TECHNOLOGY (IJMET)
Scopus Indexed Journal

www.iaeme.com/ijmet/index.asp

Certificate of Publication

This is to certify that the research paper entitled **"GRAPH-BASED MODELING AND DIJKSTRA ALGORITHM FOR SEARCHING VEHICLE ROUTES ON HIGHWAYS"** authored by **"Sodik Kirono, Muhammad Irfan Arifianto, Reza Eka Putra, Ahmad Musoleh, Roby Setiadi"** had been reviewed by the Editorial Board and published in **"International Journal of Mechanical Engineering & Technology (IJMET), Volume 9, Issue 8, August 2018, pp. 1273–1280; ISSN Print: 0976-6340 and ISSN Online: 0976-6359; Journal Impact Factor (2016): 9.2286 Calculated by GISI (www.jifactor.com); InfoBase Index IBI Factor for the year 2015–16 is 3.46; Thomson Reuters' Researcher ID: B-7384-2016"**.

Chief Editor

GRAPH-BASED MODELING AND DIJKSTRA ALGORITHM FOR SEARCHING VEHICLE ROUTES ON HIGHWAYS

Sodik Kirono, Muhammad Irfan Arifianto, Reza Eka Putra, Ahmad Musoleh
Stikom Poltek Cirebon, Cirebon, West Java 45153, Indonesia

Roby Setiadi*
Universitas Muhadi Setiabudi, Brebes, Central Java 52252, Indonesia
*Corresponding Author Email: robysetiadi.psdie.undip@gmail.com

ABSTRACT

Dynamic road conditions are one of the challenges in finding optimal vehicle routes. One of the problems that may arise is a change in road conditions that will be traversed suddenly, for example a sudden traffic jam due to an accident or an unexpected disaster. When this happens, the initial route that was previously used must be recalculated to get an alternative route before the driver enters the traffic jam. This article aims to design a graph-based model that utilizes the Dijkstra Algorithm to help motorists obtain alternative routes to go through. Graph model is certainly a very effective model for representing a model of a highway network. In addition, the Dijkstra Algorithm itself is widely known as the optimal method for finding the shortest path from the starting point to the intended endpoint.

Key words: graph, dijkstra, alternative route, shortest path, transportation.

Cite this Article: Sodik Kirono, Muhammad Irfan Arifianto, Reza Eka Putra, Ahmad Musoleh, Roby Setiadi, Graph-Based Modeling and Dijkstra Algorithm for Searching Vehicle Routes on Highways, *International Journal of Mechanical Engineering and Technology* 9(8), 2018, pp. 1273–1280.
http://www.iaeme.com/IJMET/issues.asp?JType=IJMET&VType=9&IType=8

1. INTRODUCTION

The optimal route search is one of the real-world problems that can be solved using a computer algorithm approach. To be able to do this, first the road network needs to be modeled appropriately so that computer algorithms can be applied effectively to find the route of a vehicle. Graph theory is a very appropriate approach for modeling highway networks. In graph theory, we can represent each intersection on a highway as a node or point that is connected to a node or other intersections through a highway connection, where in each connection we add a weight value that states the 'cost' of the journey from the origin node to the destination node.

In graph theory, there is also a process of finding a route or route from a starting point to an end point, where the route must have a minimum travel cost. This process is commonly referred to as the shortest path finding. There are a number of computer algorithms that can be applied to solve this problem, one of the popular and very effective in finding the optimal route is the Dijkstra Algorithm.

Scientists in the field of computer science have long conducted research to solve the problem of shortest path finding in various cases using the Dijkstra Algorithm. Some of them apply the case study to search for disaster evacuation routes [1], search for in and out routes of vehicles carrying building materials in the project area [2], search for paths to the nearest parking lot [3], and find the optimal travel route to save fuel usage and time [4-6]. Nevertheless, the Dijkstra Algorithm is not only useful for finding travel routes, but can also be applied to other cases that can be modeled into graphs [13]

This article aims to present a road network modeling scheme using graph theory, as well as provide an overview of the procedures for finding vehicle travel routes by minimizing travel costs [11-12, 14]. In this case, the cost of the trip can be in the form of vehicle distance from the point of departure to the destination point, travel costs (eg must pass the toll road), conditions of road density, road surface conditions, or even road insecurity conditions for accidents, etc. By understanding the problems to be solved, the graph model that is created will be able to provide optimal solutions.

2. RESEARCH METHODS

2.1. Graph Theory

Graph is a modeling in the form of a set of objects called points or nodes that are connected to each other through a line called a vertex. Based on the type of vertex, the graph can be divided into two types, namely directed graph and non-directed graph. The existence of a direction on a vertex that connects two nodes states that the adjacency relationship between the two codes only applies from the starting point to the end point indicated by the direction of the arrow at the vertex. On the contrary, on the non-directed graph, the relationship of the relationship applies from both sides of the node. In addition, at the vertex can also be added a value which is usually called weights to express the value of adjacency between the two nodes connected to it. Figure 1 illustrates directed and non-directed graphs.
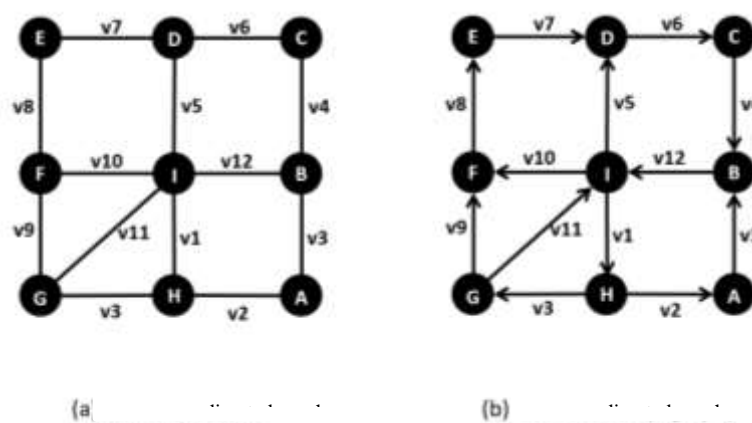
Figure 1 Illustration of non-directed graph (a) and directed graph (b). Each vertex on the graph has its own weight.

The adjacency relationship in the graph can be represented in the matrix structure as shown in Figure 2, where the value in the matrix indicates the presence or absence of relationships between nodes. A value of 1 states that there is a adjacency relationship between

two nodes, while a value of 0 states that there are no adjacency relationships between these nodes. This matrix is commonly referred to as adjacency matrix or adjacency matrix.

In computer science, graphs are often used to represent relationships between cities or regions in a region. The connection between one city and another city is indicated by the existence of a road that connects the two. Another example is to represent the road relations in an area, where each intersection will be marked as a node, while the vertex represents a road connection between the intersection of one with the other intersection.
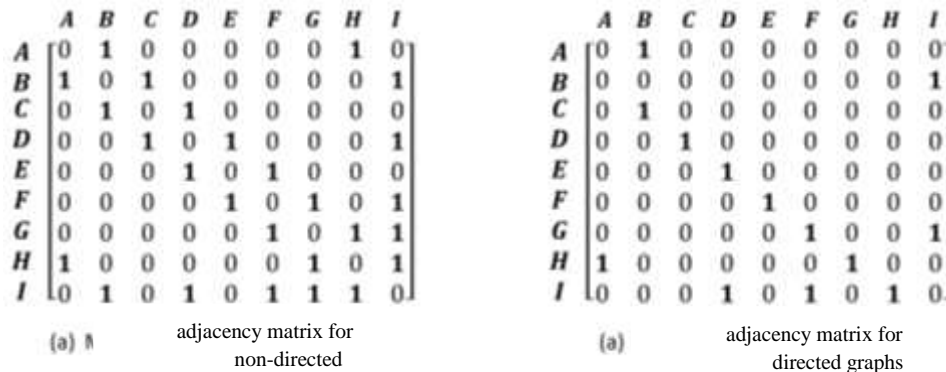
	A	B	C	D	E	F	G	H	I
A	0	1	0	0	0	0	0	1	0
B	1	0	1	0	0	0	0	0	1
C	0	1	0	1	0	0	0	0	0
D	0	0	1	0	1	0	0	0	1
E	0	0	0	1	0	1	0	0	0
F	0	0	0	0	1	0	1	0	1
G	0	0	0	0	0	1	0	1	1
H	1	0	0	0	0	0	1	0	1
I	0	1	0	1	0	1	1	1	0

(a) adjacency matrix for non-directed

	A	B	C	D	E	F	G	H	I
A	0	1	0	0	0	0	0	0	0
B	0	0	0	0	0	0	0	0	1
C	0	1	0	0	0	0	0	0	0
D	0	0	1	0	0	0	0	0	0
E	0	0	0	1	0	0	0	0	0
F	0	0	0	0	1	0	0	0	0
G	0	0	0	0	0	1	0	0	1
H	1	0	0	0	0	0	1	0	0
I	0	0	0	1	0	1	0	1	0

(a) adjacency matrix for directed graphs

Figure 2 The adjacency matrix for non-directed graphs and directed graphs in Figure 1.

Based on the adjacency matrix in Figure 2, it can be seen that the non-directed graph has a symmetrical adjacency matrix (there is a reflection between the diagonal of the matrix). This is because the relationship between node A and node B is considered the same as the relationship of node B to node A. Conversely, it is not the case with the adjacency matrix on directed graphs, because in the graph, the relationship between node A to node B is not the same as node B to node A, because the adjacency is determined by the vertex direction on both nodes.
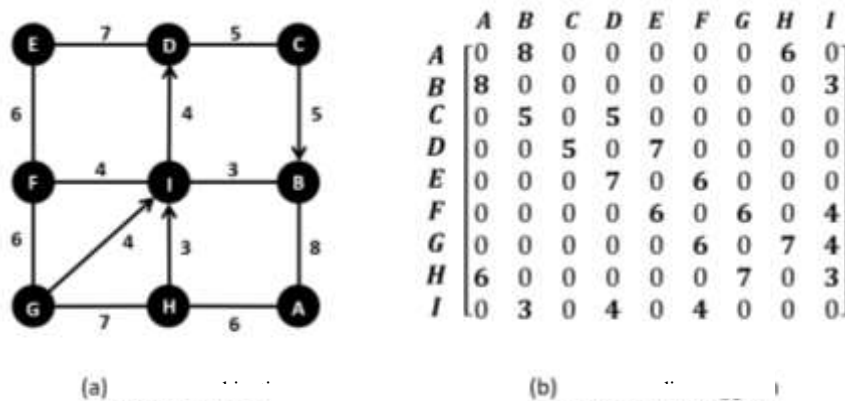

	A	B	C	D	E	F	G	H	I
A	0	8	0	0	0	0	0	6	0
B	8	0	0	0	0	0	0	0	3
C	0	5	0	5	0	0	0	0	0
D	0	0	5	0	7	0	0	0	0
E	0	0	0	7	0	6	0	0	0
F	0	0	0	0	6	0	6	0	4
G	0	0	0	0	0	6	0	7	4
H	6	0	0	0	0	0	7	0	3
I	0	3	0	4	0	4	0	0	0

(a)　　　　　　　　　　　(b)

Figure 3 Illustration of combination graph and adjacency matrix with vertex weight

In the case study of highway network modeling, the two graph models can be implemented simultaneously (combination), for example a directed graph is used to express the existence of direct current from node A to node B, while the non-directed graph is used to express the existence of two-way currents at node A and node B (see Figure 3). The weight of each vertex on the road network can also be included in the adjacency matrix by replacing the number 1 in the adjacency matrix with the weight of the vertex, where the weight of the vertex between the two nodes of the size> 0 states that there is a relationship between the two nodes with the weight contained in matrix.

2.2. Shortest Path Finding

In graph theory, the term shortest path finding is known as a route search mechanism from the starting point to the end point with minimal travel costs. As mentioned earlier that travel costs can be in the form of vehicle mileage from the point of departure to the destination point, travel costs (eg must pass through toll roads, gasoline costs), conditions of road density, road surface conditions, or even road hazard conditions to accidents, etc. In its implementation, several types of costs can be combined, so the system is representative enough to model the actual road conditions, for example the weight of the vertex is calculated based on two types of costs, namely travel distance, and road density conditions. Equation (1) aims to calculate the total weight of each vertex on a graph by combining several types of costs, where i is the index node on the graph, j is the weight type index vertex, and $w_{i,j}$ is the cost value at the i vertex, the type of weight j.

$$v_i = \sum_{j=1}^{N} w_{i,j} \tag{1}$$

Equation (1) can basically be expanded by adding a percentage for each type of cost, with the aim of giving a higher value to certain costs. For example, if the condition of road density is considered more important than the distance traveled, then the cost of road density conditions can be given a percentage value that is higher than the cost of the trip distance. This can be represented in equation (2), where p_i is the percentage for the j cost.

$$v_i = \sum_{j=1}^{N} p_j \cdot w_{i,j} \quad \text{with} \quad \sum_{j=1}^{N} p_j = 1 \tag{2}$$

2.3. Dijkstra's Algorithm

The algorithm proposed by Edger Dijkstra in 1959 was able to work effectively to find the optimal path that has the lowest total travel cost from the starting point to the intended endpoint [5]. This algorithm accepts input in the form of a weighted graph, an initial node, and an end node that is the destination of the route search. In fact, the Dijkstra Algorithm is claimed to be able to find solutions faster than some other algorithms such as A * (A star) and the Ant Colony Algorithm [7]. The Pseudocode of the Dijkstra Algorithm can be seen in Table 1.

Table 1 Pseudocode from the Dijkstra Algorithm

Algoritma Dijkstra

```
Set adjacency matrix from graph G
s = start node
g = goal node
cost = 0
bestnode = s

while bestnode ≠ g
    for all successors of bestnode do
       calculate the distance from s to these successors
    end for

    best_successor = successor with minimum distance from s
    change bestnode with this current best_successor
    update cost from s to bestnode
end while
```

In general, the work procedure of this algorithm begins with initializing the initial node (s) and the end node (g). Then, the optimal route search is done iteratively until the algorithm reaches the end node. The search process starts by listing all the nodes that are adjacent to the node s, then the adjacency node that has the total weight or the smallest total cost of travel is saved as bestnode. After that, the search process is repeated with bestnode as the initial node. All adjacency nodes with the best node are accessed and selected adjacency nodes that have the smallest total cost of travel, calculated starting from the node s. Adjacency nodes that have the smallest total cost of travel costs (starting from node s) are stored as new bestnodes. And so on until the search reaches the node g.
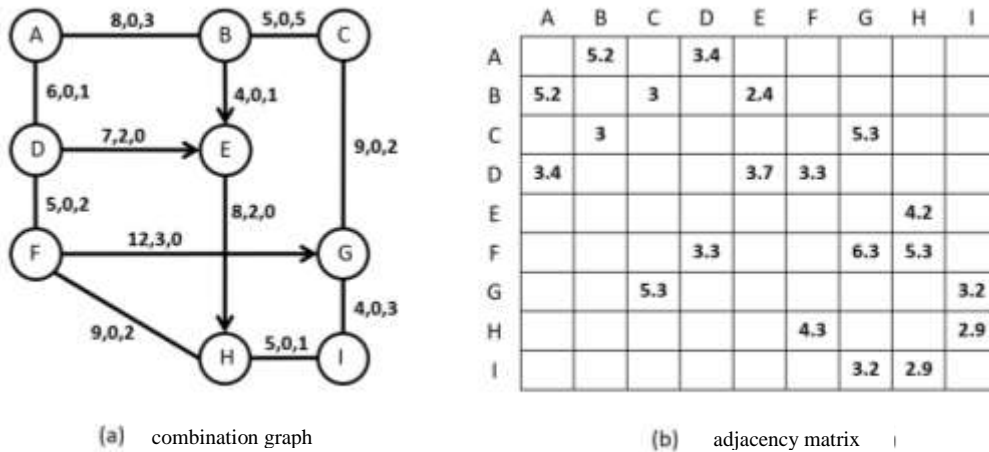
3. RESULTS AND DISCUSSION

In this article, modeling is carried out using a combination graph with three cost components, namely travel distance (w_1), travel costs / toll costs (w_2), and road surface conditions (w_3). In this modeling, the distance traveled is considered as the dominant cost factor, so the percentage will be given the highest value on the scale [0.1], while the cost of travel / toll costs is considered as a cost factor that is not too important, so that the smallest percentage is given. For example, w_1 is given a percentage of 0.5, w_2 is given a percentage of 0.1, and w_3 is given a percentage value of 0.4 (the total percentage of $w_1 + w_2 + w_3 = 1$). Thus, if translated based on equation (1) and equation (2), equation (3) is obtained.

$$v_i = \sum_{j=1}^{3} 0.5 * w_{i,1} + 0.1 * w_{i,2} + 0.4 * w_{i,3} \tag{3}$$
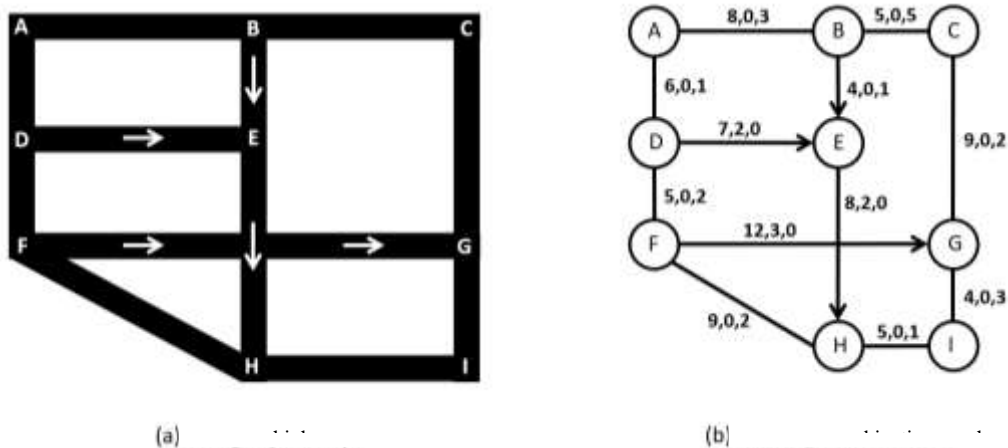
Figure 4 Modeling the road network into a combination graph model. Each intersection is converted to a node. EH and FG lanes are multilevel toll roads.

The graph model used can be seen in Figure 4, which is a combination graph with 9 nodes, where node A is the starting point and node I is the end point. Each vertex contains the values w1, w2, and w3 sequentially, for example vertices that connect nodes A and B have weights (8.0,0), which means the cost of the distance from A to B is 8, the toll cost is 0 (meaning vertex this is not a toll road), and the cost of road conditions is 3 (meaning that the road is not so smooth, some roads are hollow or uneven). The greater the value of the cost, the greater the total weight of the vertex. The value in the adjacency matrix in Figure 5 is obtained by calculating the cost at each vertex using equation (3).

	A	B	C	D	E	F	G	H	I
A		5.2		3.4					
B	5.2		3		2.4				
C		3					5.3		
D	3.4				3.7	3.3			
E								4.2	
F			3.3				6.3	5.3	
G			5.3						3.2
H						4.3			2.9
I							3.2	2.9	

(a)   combination graph   (b)   adjacency matrix

**Figure 5** Combined graph model with 9 nodes and adjacency matrix

With a model like this, the optimal route that can be obtained using the Dijkstra Algorithm is A-D-E-H-I with a total travel cost of 14.2, as shown in Figure 6. The resulting route is via two toll roads on vertices DE and EH. Both toll roads are able to minimize travel costs, because the w_3 cost component (condition of the road surface) is equal to 0, meaning that the road being passed is smooth and there is no damage to the road surface.
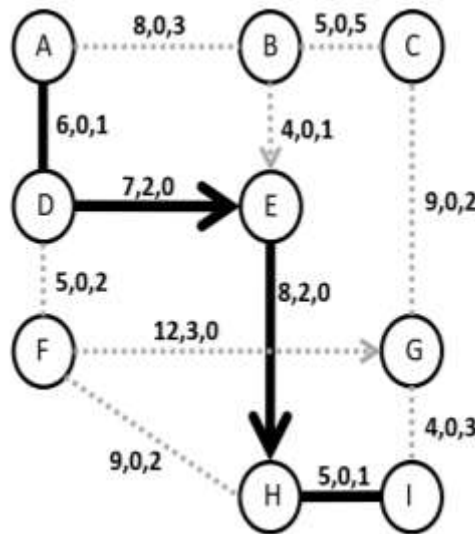
Figure 6 The optimal route obtained is A-D-E-H-I with a total cost of 14.2

Different results will be encountered if the percentage component at each vertex is omitted, meaning that each component of the travel costs is considered to have the same level of importance. In this way, the ADEHI route is no longer the optimal route, but instead changes to the ADFHI route with a total travel cost of 31 (see Figure 7), while the ADEHI route has a total cost of 32. This shows that the assignment of weight to the component of the travel costs will affect the route generated by the Dijkstra Algorithm.
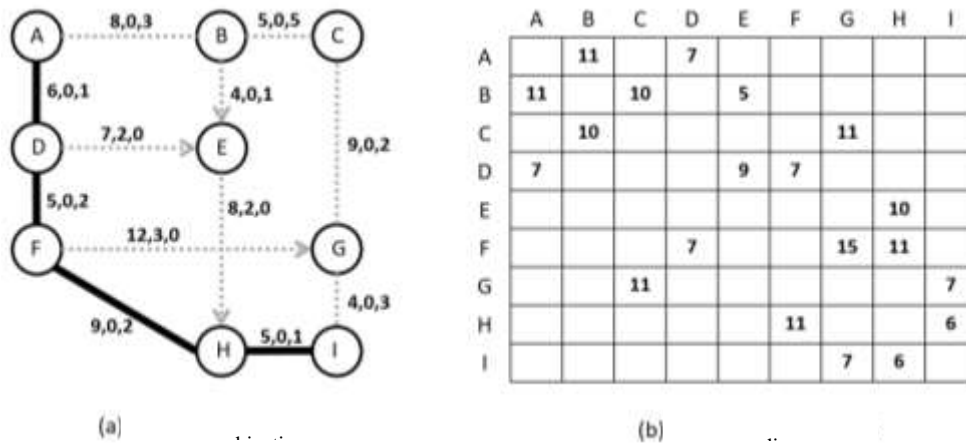
Figure 7 The optimal route generated is A-D-F-H-I if weighting vertices without a percentage component in travel costs

4. CONCLUSIONS

From the elaboration described earlier, it can be concluded that graphs can effectively model the road network, so that it can be used for computing processes. In addition, the Dijkstra Algorithm is also effective in finding the optimal route on the graph, namely the route with the smallest travel costs. The trip cost can be in the form of a single component cost, or it can be in the form of multiple components. At multiple component costs, the percentage of each component of costs affects the optimal route generated by the Dijkstra Algorithm.

REFERENCES

[1]   Miler, M., Medak, D., Odobasic, D. (2013). The Shortest Path Algorithm Performance Comparison in Graph and Relational Database on a Transportation Network. *Information and Communication Technology Preliminari Communication*, 26(1), 75-82.

[2]   Solthani, A.R., Tawfik, J.Y., Goulermas., Fernando, T. (2002). Path Planning in Construction Sites: Performance Evaluation of the Dijkstra, A*, and GA search algorithm, *Advanced engineering informatics 16,* 291-303.

[3]   Jaafar, H., Zabidi, M.H., Soh, A.C., Hoong, T.P., Shafie, S., and Ahmad, S.A. (2014). Intelligent Guidance Parking System using Modified Dijkstra Algorithm. *Journal of Science and Technology*, 132-141.

[4]   Alazab, A., Venkatraman, S., Abawajy, J., Alazab, M. (2011). *An Optimal Transportation Routing Approach using GIS-based Dynamic Traffic Flows*. Int'l. Conf. on Info. And Financial Eng., 172 – 178.

[5]   Haixiang, D. and Jingjing, T. (2013). The Improved Shortest Path Algorithm and Its Application in Campus Geographic Information Systems. *Journal of Convergence Information Technology*, 8(2). doi: 10.4156/jcit.vol8.issue2.5

[6]   Amaliah, B., Fatichah, C., and Riptianingdyah, O. (2016). Finding the Shortest Paths among Cities in Java Island using Node Combination based on Dijkstra Algorithm. *Int'l Journal on Smart Sensing and Intellligent Systems*, 9(4), 2219-2236.

[7]   Choubey, N., Bhupesh., and Gupta, K.R. (2013). Analysis of Working of Dijkstra and A* to Obtain Optimal Path. *International Journal of Computer Science and Management Research,* 2(3).

[8]   Mulyawan, H. (2018). Functions of Financial Services Authority in Supervision of Auto Insurance Companies. *Asian Journal of Law and Jurisprudence 1*(1).

[9]   Prayogo, G. (2018). Bitcoin, Regulation and the importance of National Legal Reform. *Asian Journal of Law and Jurisprudence 1*(1), 1-9.

[10]   Wirawan, A. (2018). Future Direction of the Politics of Law of the Public Service. *Asian Journal of Law and Jurisprudence 1*(1), 17-22.

[11]   Bachri, O.S., Subagio, A., Rasta & Budiatmaja, B. S. (2018). Development of geographic information system for searching shortest trip route on online taxi. *International Journal of Mechanical Engineering and Technology 9*(7), 1045-1050.

[12]   Bachri, O. S., Kusnadi, Hatta, M., Nurhayati, O. D. (2017). Feature selection based on chi square in artificial neural network to predict the accuracy of student study period. *International Journal of Civil Engineering and Technology 8*(8), 731 – 739

[13]   Fitro, A., Bachri, O.S., Sulistio Purnomo, A.I., & Frendianata, I. (2018). Shortest path finding in geographical information systems using node combination and dijkstra algorithm. *International Journal of Mechanical Engineering and Technology 9*(2), 755-760.

[14]   Nurhayati, O. D., Bachri, O. S. Supriyanto, A., & Hasbullah, M. (2018). Graduation Prediction System Using Artificial Neural Network. *International Journal of Mechanical Engineering and Technology 9*(7), 1051-1057