Lebesgue: Jurnal Ilmiah Pendidikan Matematika, Matematika dan Statistika p-ISSN 2721-8929
http://lebesgue.lppmbinabangsa.id/index.php/home e-ISSN 2721-8937
Vol. 4, No. 2, Agustus 2023 DOI Issue: 10.46306/lb.v4i2

DIJKSTRA ALGORITHM IN DETERMINING THE SHORTEST ROUTE FOR DELIVERY SERVICE BY J&T EXPRESS IN BANDUNG

Anie Lusiani[1,*] Siti Samsiyah Purwaningsih[2] Euis Sartika[3]
[1]Department of Mechanical Engineering, Politeknik Negeri Bandung
[2,3]Department of Commercial Administration, Politeknik Negeri Bandung
Email: anie.lusiani@polban.ac.id.

ABSTRACT

The determination of the shortest route can be done using various methods, one of which is the Dijkstra algorithm. This algorithm is often used in routing problems with minimum weight in computer networks, communication networks and transportation networks. There are several applications of the Djikstra algorithm, namely solving the problem of the shortest path of neutrosophy, mapping tourism potential, and developing geographic information systems in the health sector. In this research, Dijkstra algorithm will be applied to optimize the mileage of the delivery of goods packages. Data and information namely the address, weight and number of customer packages carried by a courier in one delivery trip are obtained from the drop point of PT. J&T Express in Sarijadi area of Bandung City. Meanwhile, data about mileage is obtained from the Google Maps application. All this data is used to construct an initial model graph that is a connected weighted graph, where the location of a drop point or a customer is a vertex and a road connecting two locations is an edge on the graph. The weight in this graph is the mileage from the drop point to the customers or from one customer to another. Then, the Djikstra algorithm is run on this graph where the drop point is the starting point of the route, so that the courier visits all customers and returns again to the drop point. The resulting route is a cycle in the graph which is the shortest closed route at 1890 meters.

Keywords: *connected weighted graph, Dijkstra algorithm, J&T Express, shortest route*

Received: July 05, 2023 / Accepted: August 01, 2023 / Published Online: August 30, 2023

Vol. 4, No. 2, Agustus 2023 hal. 940-948
DOI Artikel: 10.46306/lb.v4i2.337

INTRODUCTION

In general, the application of the Dijkstra algorithm is optimization in the problems of computer networks, communication networks, and transport networks. Theoretically, the Dijkstra algorithm is an algorithm used in determining routes in networks with minimum weight. This weight is the travel time, mileage, or cost needed during the trip through a route. This algorithm was found by Edsger W. Dijkstra in 1956 (Ahmed, Ahmed, and Ahmed 2017). Thus, the Dijkstra algorithm can be used for the application of graph theory in the determination of the shortest route in the network of the package delivery service companies. The shortest route is generated by running the Dijkstra algorithm on the initial model graph, starting from the starting point to all other points and returning again to the starting point. Thus, this shortest route is a closed route traversed by a courier from a package delivery service company in a trip. The graphs used in this study are simple and finite graphs. A graph that is connected and its edges have a weight is called a connected weighted graph. This initial model graph is a connected weighted graph, where the weight is the mileage. We call a point in a graph a vertex. So, the starting point is the starting vertex. The shortest closed route generated from this algorithm can be either a cycle or a circuit. A cycle is a closed route that visits a vertex once, otherwise it is a circuit. For example, $C_1 = (2, 3, 4, 2)$ is a cycle, but $C_2 = (1, 2, 3, 4, 2, 3, 5, 1)$ is a circuit in a graph, where $\{1, 2, 3, 4, 5\}$ is a vertex set of this graph (Chartrand, Lesniak, and Zhang 2016).

Optimal distribution management capabilities will have a major impact on all aspects, especially for freight package delivery service companies. Recently, there have been several studies related to delivery service optimization, namely the problem of routing vehicles with drones for delivery services using ant colony optimization algorithms (Huang et al. 2022), integrating clustering methodologies and routing optimization algorithms for last-mile parcel delivery (Ramírez-Villamil et al. 2022), multi-objective optimization of bicycle routes for last-mile package delivery with drop-off (Osaba et al. 2018), synchronized truck and drone routing in package delivery logistics (Das et al. 2020), and a multi-objective optimization approach to package delivery by the crowd of occupied taxis (Zhou et al. 2022).

Optimization can also use Prim algorithm to design autonomous drone controls in packet delivery (Wirabudi, Hafiza, and Fachrurrozi 2022) and determine the shortest or fastest path in logistics distribution (Lusiani, Sartika, Habinuddin, et al. 2021). The Prim algorithm can be compared to the Floyd-Warsall algorithm in optimization to determine the shortest path (Ramadhan, Siahaan, and Mesran 2018). Using a methodology based on the Prim Algorithm, we can improve the reliability index using artificial immune system techniques by applying graph theory considerations to improve computational performance and Pareto's rule of dominance (Alonso, Oliveira, and De Souza 2014).

As an application in graph theory, a study has been conducted by Ginting et al. in 2019 on delivery simulation using the Dijkstra algorithm to solve the problem of traveling salesmen (Ginting, Osmond, and Aditsania 2019). Dijkstra algorithm can also be used to determine current policies and calculate IRL (inverse reinforcement learning) gradients in considering the characteristics of food delivery routes (Liu et al. 2020), determine the fastest path in logistics distribution by Bulog in the West Java region (Lusiani, Sartika, Binarto, et al. 2021), and solve the shortest path problem with fuzzy arc lengths (Deng et al. 2012). The application of graphs as models to a problem is not only in determining the shortest path and the fastest path. A graf is also used as a model on traffic congestion problems, such as determining waiting times in overcoming congestion at the intersection of Toll-Pasteur Bandung road (Lusiani et al. 2020). In this article, we will determine the shortest route for delivery service by PT. J&T Express in Bandung using Dijkstra algorithm.

METHOD

The model to be analyzed is the case of a package delivery service carried out by a courier from a drop point of PT. J&T Express which is located at Jl. Sarijadi Raya No. 90, Sukarasa, Kec. Sukasari, Bandung City. The number of customers served by a courier is more than 15 customers in one delivery trip depending on the density of demand. In this case, we will analysis for 17 customers in 12 locations in the Gegerkalong Hilir and Sarijadi areas. Meanwhile, data about mileage is obtained from the Google Maps application. All this data is used to construct an initial model graph which is a connected weighted graph, where the location of the drop point or a customer is a vertex and the road connecting two locations is an edge of the graph. The weight in this graph is the distance traveled from the drop point to the customers or from one customer to another. We call the distance traveled is the mileage. Then, the Djikstra algorithm is run on this graph where the drop point is the starting point of the route, such that the courier visits all customers and returns again to the drop point.

The flow of research methodology can be described as follows. First, collecting of customer location/address data. Then, processing location data into distance data traveled from one location to another. Second, constructing this data as an initial model i.e. a connected weighted graph. Third, running the Dijkstra algorithm on this graph so that the shortest route is obtained. This shortest route should visit all customers and return again to the drop point. Thus, this route is either a closed path (cycle) or a closed trail (circuit) that is a subgraph of the initial model graph.

The initial model graph is constructed from data on the number of customers, customer locations, the existence of roads connecting one location to another, road conditions traveled, and mileage between locations. The road connecting these locations must be passable by a motorcycle. In this model, if there is no road connecting 2 locations directly then there is no edge between the 2 locations. Thus, the initial model graph of this algorithm is a connected weighted graph, namely *G*, as shown in Figure 1.

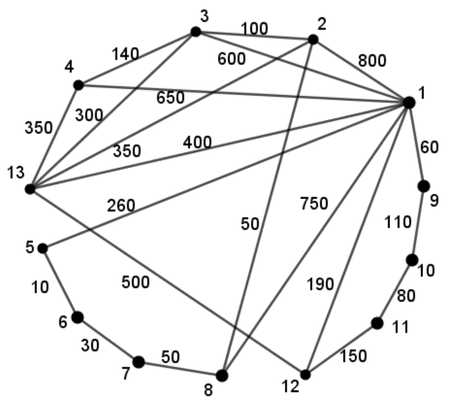

Figure 1 The Initial Model Graph *G*

Vertex 1 is the drop point of PT. J&T Express Sarijadi which is the starting point or the initial vertex of the route. Vertex 2 until vertex 13 are customers who will be visited by a courier one by one sequentially according to the Dijkstra algorithm. The initial model constructed graph is not a complete graph because not all vertices are directly connected to other vertices by an edge. It is a graph with 13 vertices and 21 edges.

Figure 2 shows a matrix adjacent of the initial model graph. Matrix elements are weights on the edges of the graph. The weight of the edges is the mileage from one location to another in meters. The element on the main diagonal of the matrix is zero, since there is

Vol. 4, No. 2, Agustus 2023 hal. 940-948
DOI Artikel: 10.46306/lb.v4i2.337

no distance between the same vertices. This matrix is symmetrical, because the mileage from vertex *i* to vertex *j* is equal to the mileage from vertex *j* to vertex *i*.

$$\begin{pmatrix}
0 & 800 & 600 & 650 & 260 & 0 & 0 & 750 & 60 & 0 & 0 & 190 & 400\\
800 & 0 & 100 & 0 & 0 & 0 & 0 & 50 & 0 & 0 & 0 & 0 & 350\\
600 & 100 & 0 & 140 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 300\\
650 & 0 & 140 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 350\\
260 & 0 & 0 & 0 & 0 & 10 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 10 & 0 & 30 & 0 & 0 & 0 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 30 & 0 & 50 & 0 & 0 & 0 & 0 & 0\\
750 & 50 & 0 & 0 & 0 & 0 & 50 & 0 & 0 & 0 & 0 & 0 & 0\\
60 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 110 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 110 & 0 & 80 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 80 & 0 & 150 & 0\\
190 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 150 & 0 & 500\\
400 & 350 & 300 & 350 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 500 & 0
\end{pmatrix}$$

Figure 2 Matrix Adjacent of Graph *G*

The steps of the Dijkstra algorithm work as follows. The first step is to establish a starting point with the status of having been found and has been visited. A courier picks up packages of goods sent to customers at the Sarijadi drop point, so vertex 1 is the starting point of the route. Thus, the status of vertex 1 has been found and visited. The second step is to find other vertices that are adjacent to vertex 1. If these vertices have not yet found, then change their status to be found. However, if it has been found then only update the weight. The third step is to choose a vertex with a minimum weight to visit.

This second and third steps result in a vertex always being found, if the vertex is connected to a vertex that has been visited (in fact, a vertex is found more than once if the vertex is connected to some vertices that have been visited). But not all vertex needs to be visited. In the optimization process, if a vertex does not have a minimum weight, then the vertex does not need to be visited. Next, we repeat the second and third steps, so that all vertices have been visited and the courier returns to the initial vertex (vertex 1). The following figure and table show the second and third steps in the first loop on Dijkstra algorithm.

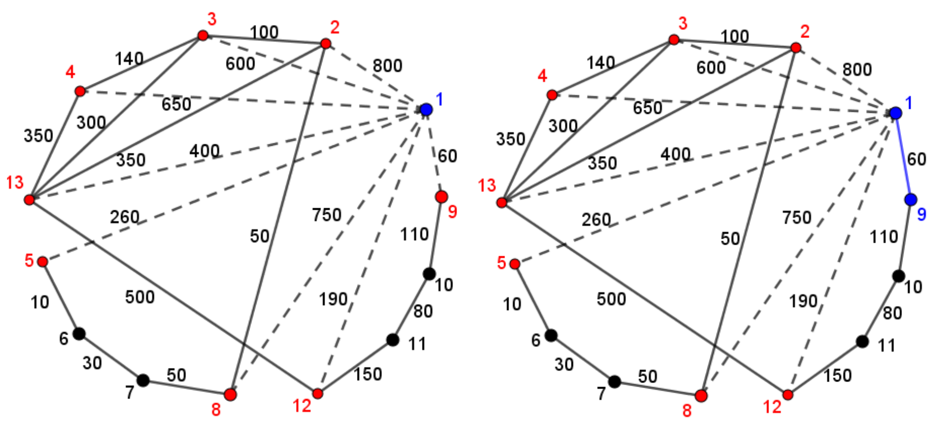

Figure 3 Graph *G* in The First Loop of Dijkstra Algorithm

Vertex 1 has been determined as the starting point, so in Figure 3, vertex 1 is coloured blue to indicate it has been found and visited. The first row in Table 2 shows the route from vertex 1 to vertex 1 yields zero weight. In the second step, vertices 2, 3, 4, 5, 8, 9, 12, and 13 are adjacent to vertex 1. Then, these vertices are coloured by red (see Figure 3 on the left). Since {1, 9} is the edge with minimum weight of 60 meters, the next vertex to visit is vertex 9. So, vertex 9 is coloured by blue (see Figure 3 on the right). This is also shown in the ninth row of Table 2.

Table 2 The Vertices Conditions in The First Loop of Algorithm Dijkstra

Vertex	Status	Weight	Route
1	visited	0	1
2	found	800	(1, 2)
3	found	600	(1, 3)
4	found	650	(1, 4)
5	found	260	(1, 5)
6	not yet found	∞	-
7	not yet found	∞	-
8	found	750	(1, 8)
9	visited	60	(1, 9)
10	not yet found	∞	-
11	not yet found	∞	-
12	found	190	(1, 12)
13	found	400	(1, 13)

The next step is to loop sequentially from second step to third step until the all of vertex has been visited and the courier returns again to vertex 1 as the starting point. The following figure shows the second and third steps in the fourth loop on Dijkstra algorithm.

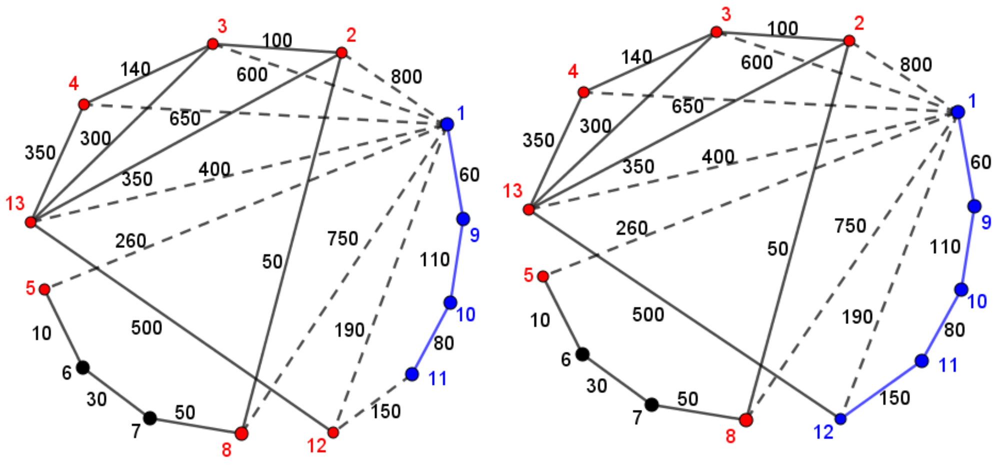

Figure 4 Graph *G* in The Fourth Loop of Dijkstra Algorithm

Figure 4 on the left shows vertices 9, 10, and 11 have been found and visited, so these vertices are coloured by blue. While, vertex 12 has been found and coloured by red. Then, Figure 4 on the right shows vertex 12 has been visited, since vertex 12 is the only vertex found by vertex 11, so vertex 12 is coloured by blue.

Table 3 The Vertices Conditions in The Fourth Loop of Dijkstra Algorithm

Vol. 4, No. 2, Agustus 2023 hal. 940-948
DOI Artikel: 10.46306/lb.v4i2.337

Vertex	Status	Weight	Route
1	visited	0	1
2	found	800	$(1,2)$
3	found	600	$(1,3)$
4	found	650	$(1,4)$
5	found	260	$(1,5)$
6	not yet found	∞	-
7	not yet found	∞	-
8	found	750	$(1,8)$
9	visited	60	$(1,9)$
10	visited	170	$(1,9,10)$
11	visited	250	$(1,9,10,11)$
12	visited	400	$(1,9,10,11,12)$
13	found	400	$(1,13)$

The ninth until the twelfth rows in Table 3 show vertices 9, 10, 11, and 12 have been found and visited, vertices 2, 3, 4, 5, 8, and 13 have been found, while vertices 6 and 7 have not found. The route from vertex 1 to vertex 12 is $(1,9,10,11,12)$ for a total weight of 400 meters.

RESULTS AND DISCUSSION

In the fifth loop, we have found vertices 1 and 13, because vertex 12 adjacent to these vertices. But, vertex 1 has been visited. So, we must chose vertex 13 to visit. Since vertices 4, 3, and 2 are close together, we chose these vertices to visit sequentially on the 6th through 9th loops. Therefore, these three vertices have a status of having been visited and are coloured by blue. The route from vertex 1 to vertex 2 is $(1,9,10,11,12,13,4,3,2)$ for a total weight of 1490 meters.

We find out some vertices that have not yet visited, after visit vertex 2. Since vertex 2 is adjacent to vertices 1, 3, 8 and 13, so we find vertex 8. Since vertices 5, 6, 7, and 8 are close together, we chose these vertices to visit sequentially until on the final loops. Thus, the last customer to visit on this trip is vertex 5.

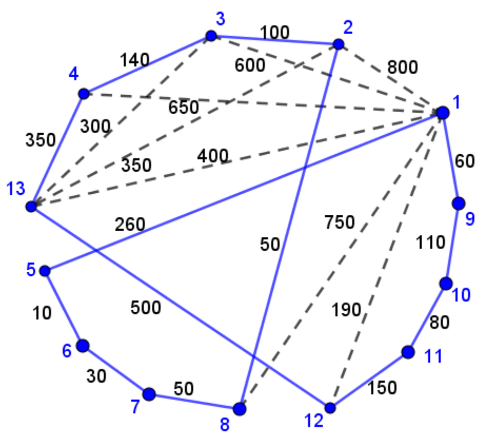

Figure 5 Graph *G* in The Last Loop of Algorithm Dijkstra

Next, the courier must return to the starting point, vertex 1, to report the delivery results. This step from vertex 5 to vertex 1 is the last loop because there is only one road that connects directly between these vertices. Therefore, the final route at this trip as the

Vol. 4, No. 2, Agustus 2023 hal. 940-948
DOI Artikel: 10.46306/lb.v4i2.337

shortest closed route as shown in Figure 5 is a cycle $(1, 9, 10, 11, 12, 13, 4, 3, 2, 8, 7, 6, 5, 1)$ with a weight of 1890 meters as shown in the following table.

Table 5 The Vertices Conditions in The Last Loop of Algorithm Dijkstra

Vertex	Status	Weight	Route
1	visited	1890	$(1, 9, 10, 11, 12, 13, 4, 3, 2, 8,7,6,5,1)$
2	visited	1490	$(1, 9, 10, 11, 12, 13, 4, 3, 2)$
3	visited	1390	$(1, 9, 10, 11, 12, 13, 4, 3)$
4	visited	1250	$(1, 9, 10, 11, 12, 13, 4)$
5	visited	1630	$(1, 9, 10, 11, 12, 13, 4, 3, 2, 8,7,6,5)$
6	visited	1620	$(1, 9, 10, 11, 12, 13, 4, 3, 2, 8,7,6)$
7	visited	1590	$(1, 9, 10, 11, 12, 13, 4, 3, 2, 8,7)$
8	visited	1540	$(1, 9, 10, 11, 12, 13, 4, 3, 2, 8)$
9	visited	60	$(1, 9)$
10	visited	170	$(1, 9, 10)$
11	visited	250	$(1, 9, 10, 11)$
12	visited	400	$(1, 9, 10, 11, 12)$
13	visited	900	$(1, 9, 10, 11, 12, 13)$

CONCLUSION

After running Dijkstra algorithm on the initial model graph G, the shortest closed route as a subgraph of connected weighted graph G is obtained. This route is a cycle $(1, 9, 10, 11, 12, 13, 4, 3, 2, 8, 7, 6, 5, 1)$ with a weight of 1890 meters. This route is a cycle that containing every vertex in graph G. A graph of order n is called Hamiltonian if it contains a cycle of length n (Lusiani, Baskoro, and Saputro 2017). The number of vertex in graph G is 13 and the length of cycle $(1, 9, 10, 11, 12, 13, 4, 3, 2, 8, 7, 6, 5, 1)$ is 13. Therefore, graph G is a Hamiltonian.

For different cases, it is possible to find a closed route that is a circuit. Thus this route is a closed trail, so the graph G is an Eulerian. In different optimization problems, Neumann has shown that a well-known EA variant (1+1) working on encoding important permutations can find Eulerian tours of Eulerian graphs in the expected polynomial time (Neumann 2004).

If we pay attention to road direction engineering carried out by traffic police in the Sarijadi area, then the graph G will become a digraph (directed graph), which is a graph that has directions. In this case, it is possible that a road can only be traveled in one direction, so the optimization results by the Dijkstra algorithm will be different.

ACKNOWLEDGMENT

This research was supported by "DIPA POLBAN in Research scheme Penelitian Terapan (PT) Nomor: B/92.35/PL1.R7/PG.00.03/2023, Politeknik Negeri Bandung, Indonesia".

REFERENCES

Ahmed, Aram M, Sarkar Hasan Ahmed, and Omed Hassan Ahmed. 2017. "Dijkstra Algorithm Applied: Design and Implementation of a Framework to Find Nearest Hotels and Booking Systems in Iraqi." In *2017 International Conference on Current Research in Computer Science and Information Technology (ICCIT)*, IEEE, 126–32.

Alonso, F R, Denisson Q Oliveira, and A C Zambroni De Souza. 2014. "Artificial Immune

Vol. 4, No. 2, Agustus 2023 hal. 940-948
DOI Artikel: 10.46306/lb.v4i2.337

Systems Optimization Approach for Multiobjective Distribution System Reconfiguration." *IEEE Transactions on Power Systems* 30(2): 840–47.

Chartrand, Gary, Linda Lesniak, and Ping Zhang. 2016. *Graphs & Digraphs. Textbooks in Mathematics*. CRC Press, Boca Raton, FL.

Das, Dyutimoy Nirupam, Rohan Sewani, Junwei Wang, and Manoj Kumar Tiwari. 2020. "Synchronized Truck and Drone Routing in Package Delivery Logistics." *IEEE Transactions on Intelligent Transportation Systems* 22(9): 5772–82.

Deng, Yong, Yuxin Chen, Yajuan Zhang, and Sankaran Mahadevan. 2012. "Fuzzy Dijkstra Algorithm for Shortest Path Problem under Uncertain Environment." *Applied Soft Computing* 12(3): 1231–37.

Ginting, Hagai Nuansa, Andrew Brian Osmond, and Annisa Aditsania. 2019. "Item Delivery Simulation Using Dijkstra Algorithm for Solving Traveling Salesman Problem." In *Journal of Physics: Conference Series*, IOP Publishing, 12068.

Huang, Shan-Huen, Ying-Hua Huang, Carola A Blazquez, and Chia-Yi Chen. 2022. "Solving the Vehicle Routing Problem with Drone for Delivery Services Using an Ant Colony Optimization Algorithm." *Advanced Engineering Informatics* 51: 101536.

Liu, Shan et al. 2020. "Integrating Dijkstra's Algorithm into Deep Inverse Reinforcement Learning for Food Delivery Route Planning." *Transportation Research Part E: Logistics and Transportation Review* 142: 102070.

Lusiani, Anie, Euis Sartika, Endang Habinuddin, et al. 2021. "Algoritma Prim Dalam Penentuan Lintasan Terpendek Dan Lintasan Tercepat Pada Pendistribusian Logistik Bulog Jawa Barat." In *Prosiding Industrial Research Workshop and National Seminar*, , 673–77.

Lusiani, Anie, Euis Sartika, Agus Binarto, et al. 2021. "Determination of the Fastest Path on Logistics Distribution by Using Dijkstra Algorithm." In *2nd International Seminar of Science and Applied Technology (ISSAT 2021)*, Atlantis Press, 246–50.

Lusiani, Anie, Edy Tri Baskoro, and Suhadi Wido Saputro. 2017. "On Size Multipartite Ramsey Numbers for Stars versus Paths and Cycles." *Electronic Journal of Graph Theory and Applications* 5(1): 43–50. http://www.ejgta.org/index.php/ejgta/article/view/318.

Lusiani, Anie, Euis Sartika, Agus Binarto, and Endang Habinuddin. 2020. "Compatible Graphs on Traffic Lights Waiting Time Optimization." In *Proceedings of the International Seminar of Science and Applied Technology (ISSAT 2020)*, Paris, France: Atlantis Press, 467–71. https://www.atlantis-press.com/article/125949846.

Neumann, Frank. 2004. "Expected Runtimes of Evolutionary Algorithms for the Eulerian Cycle Problem." In *Proceedings of the 2004 Congress on Evolutionary Computation (IEEE Cat. No. 04TH8753)*, IEEE, 904–10.

Osaba, Eneko et al. 2018. "Multi-Objective Optimization of Bike Routes for Last-Mile Package Delivery with Drop-Offs." In *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, IEEE, 865–70.

Ramadhan, Zuhri, A Putera Utama Siahaan, and M Mesran. 2018. "Prim and Floyd-Warshall Comparative Algorithms in Shortest Path Problem." In *Proceedings of the Joint Workshop KO2PI and The 1st International Conference on Advance & Scientific Innovation*, , 47–58.

Ramírez-Villamil, Angie et al. 2022. "Integrating Clustering Methodologies and Routing Optimization Algorithms for Last-Mile Parcel Delivery." In *Computational Logistics: 13th International Conference, ICCL 2022, Barcelona, Spain, September 21–23, 2022, Proceedings*, Springer, 275–87.

Wirabudi, Andri Agustav, Lia Hafiza, and Nurwan Reza Fachrurrozi. 2022. "Design

Vol. 4, No. 2, Agustus 2023 hal. 940-948
DOI Artikel: 10.46306/lb.v4i2.337

Autonomous Drone Control For Delivery Package Using Prim Algorithm and Waypoint Method." In *2022 13th International Conference on Information and Communication Technology Convergence (ICTC)*, IEEE, 1183–88.

Zhou, Zhifeng, Rong Chen, Jian Gao, and Hu Xing. 2022. "A Multi-Objective Optimization Approach to Package Delivery by the Crowd of Occupied Taxis." *Knowledge and Information Systems* 64(10): 2713–36.