A Comparison of Ant Colony Optimization and Depth First Search for Solving Unmanned Aerial Vehicle – Ground Vehicle Routing Problem in Humanitarian Logistics

Alfredo Aryasena
Department of Mechanical and Industrial Engineering
Faculty of Engineering
Universitas Gadjah Mada
Yogyakarta, Indonesia
alfredoaryasena469947@mail.ugm.ac.id

Bertha Maya Sopha
Department of Mechanical and Industrial Engineering
Faculty of Engineering
Universitas Gadjah Mada
Yogyakarta, Indonesia
bertha_sopha@ugm.ac.id

Abstract

Time is of the essence in disaster relief operation. The casualties might be huge if disaster relief operation can't be done quickly. The broken infrastructure and post-disaster environment make it difficult to search for survivors. To solve this, Unmanned Aerial Vehicle (UAV) is deployed since UAV can survey the location easily from the air. One downside of UAV is its limited energy capacity. UAV needs to be deployed with a Ground Vehicle (GV) who charges or swaps UAV battery during the operation. As such, the coordination between UAV and GV becomes an important part in the operation. This research focuses on constructing a route for the UAV – GV to find survivors using Ant Colony Optimization (ACO). The studied case is the 2010 Merapi eruption, which is one of the volcano eruptions in Indonesia. The result of this research will be compared with the result from previous research that used Depth-First Search (DFS). The route from ACO needs 90.48 minutes to complete whereas the route from previous research needs 95.48 minutes, implying that the route evaluated using ACO results in 5 minutes faster than that constructed using DFS.

Keywords
Routing Problem, Unmanned Aerial Vehicle, Ant Colony Optimization, Depth First Search, Humanitarian Logistics

1. Introduction

According to the Badan Nasional Penanggulangan Bencana (BNPB), a National Agency for Disaster Countermeasure in Indonesia, a disaster is an event or series of events that threatens and disrupts people's lives and livelihoods caused by both natural and/or non-natural factors as well as human factors, resulting in human casualties, environmental damage, property losses, and psychological impacts. Disasters can also cause residents to evacuate because their homes have been damaged due to the disaster. Indonesia is a country prone to disasters. According to BNPB, in the period from January 1 to January 18, 2021, there have been 154 natural disasters in Indonesia, mostly floods, hurricanes and landslides. Natural disasters that occurred resulted in damaged infrastructure so that the disaster management process took a long time. One way that can be used to reduce the time is to use an Unmanned Aerial Vehicle (UAV).

The UAV is able to complete tasks 50.4% faster or more than half of the conventional methods required according to a survey by Volkmann in 2017. This proves that the UAV is suitable for tasks with limited time. This applies to post-disaster activities, such as searching for survivors and damaged infrastructure. UAVs also have some drawbacks. One of them is that the UAV has a short flight time due to energy limitations. To solve this problem, the UAV is escorted by a Ground Vehicle (GV). The GV is tasked with exchanging or recharging the UAV battery when it runs out of energy. UAV and GV coordination have been studied in several research (Luo et al., 2017; Yu et al., 2017; Ha et al., 2018). And research in Indonesia has been carried out by Larasati (2019) where UAVs are used to search for survivors of natural disasters. The method used in the Larasati (2019) research to determine UAV and GV routes is Depth-First Search.

The Depth-First Search method is a heuristic method used for graph exploration. This method starts from a node and then traces as far as possible on one branch until it finally returns and traces another branch. This tracing is repeated until all nodes have been visited. Backtracking causes the route that is built often not optimal because of the time it takes for backtracking.

1.1 Objectives

Based on the background that has been prepared, this study aims to implement Ant Colony Optimization to solve the UAV and GV routing problem and compare the performance of Ant Colony Optimization and Depth First Search. Few studies such as Asih et al. (2017) has compared various metaheuristics approach to solve routing problem in actual settings. This includes creating a routing program using Python and comparing routes from Ant Colony Optimization and Depth First Search. The case used in this study is the 2010 Merapi Disaster.

2. Literature Review

An unmanned aerial vehicle (UAV) commonly known as a drone is an aerial vehicle without a pilot, crew, and passengers. UAV is part of the unmanned aerial system which consists of a control system from the ground and a communication system with the UAV. UAV flight can be performed manually by humans, assisted by an autopilot system in various degrees, or completely on an autopilot system. UAVs have been used to complete tasks such as surveillance and reconnaissance (Manyam et al., 2017), freight forwarding (Ha et al., 2015; Dorling et al., 2017), and disaster management (American Red Cross, 2015; Chowdhury et al., 2017). In disaster management, UAVs can be used in several ways, such as detecting collapsed buildings (Hua et al., 2016), area mapping (Cahyono and Zayd, 2018), site surveys for disaster management facilities (Silva et al., 2017) and monitoring the risk of occurrence. disaster (Gomez and Purdie, 2016).

UAV flight routes are usually planned so that the UAV can work efficiently due to energy constraints. There are several studies where the determination of the UAV route is carried out by different methods. Yakici's research (2016) determines UAV flight routes for the Navy in the Prize Collecting Location and Routing Problem (PCLRP). This study evaluates the performance of the UAV by collecting point values at nodes visited. The objective of this method is to maximize the point value at each node. Ant Colony Optimization is used to solve the problem. ACO was chosen because it provides a better solution in less time. Research by Dorling et al. (2017) used another metaheuristic method, namely simulated annealing to determine routes. This study uses a cost function to determine whether the route found is good or not by calculating the energy used, battery capacity, and weight of the load lifted by the UAV. Manyam et al. (2017) investigated route determination for UAVs in surveillance & reconnaissance missions. The purpose of this research is to minimize the time to complete all given tasks. The study has stations where UAVs land and take off. However, to increase efficiency, the station must be mobile, such as an unmanned ground vehicle (Yu et al., 2017) or a mode of transportation such as a Ground Vehicle (GV) or land vehicle. With mobile stations, the coverage area of drones also increases.

In post-disaster situations, UAVs can be used for Search and Rescue (SAR) missions. UAVs can be an effective system for locating people in difficult-to-access environments (Półka et al., 2017). However, UAVs have limited energy. These limitations make the UAV only able to fly at close range. Due to these limitations, recent studies have combined UAVs and land vehicles (GVs) such as GVs (Luo et al., 2017; Luo et al., 2018). Ground vehicles serve as UAV landing stations; to recharge the battery (Yu et al., 2017) or swap the UAV battery when the GV charges another battery (Luo et al., 2018). The application of UAVs has been researched globally. For example, the problem of determining routes for UAVs with environmental constraints such as mountains (Li et al., 2018). Larasati (2019) deployed Depth-First Search which is an algorithm for tracing the graph and doing backtracking if it reaches a dead end to determine the UAV and GV routes . This can make tracing less efficient due to backtracking. For this reason, this research wants to find out whether the Depth-First Search is good enough for determining UAV and GV routes by comparing them with the Ant Colony Optimization.

3. Methods

The focus of this research is to create a drone routing program with Ant Colony System. In this study, the object of research used was data from the 2010 Merapi eruption which was used in the Larasati (2019). This data consists of the target location and the location of the UAV and GV meeting. In addition, there are secondary data obtained from literature studies such as UAV energy limits. The methodology can be broken down into three steps.

Step 1: Creating Basic ACS Program and Verification. This program implements the ACS algorithm from the research of Dorigo et al. (1997) used to solve Traveling Salesman Problem.

Step 2: Creating a small model to verify the program. This step ensures that the program is working as intended and produce a proper result.

Step 3: Running the program with the data from Merapi and compare the results with the results from Larasati (2019).

3.1 Ant Colony System

Ant Colony System is a metaheuristic method developed by Dorigo et al in 1997. Figure-1 shows the pseudocode for ACS. Each ant constructs a route by selecting a city based on the state transition rule; Ants will choose randomly between *exploration* and *exploitation*. If the ants choose *exploitation,* it will a choose with the highest pheromones. If it chooses *exploration,* it chooses a node based on this mathematical formula.

$$p_k(r,s) = \begin{cases} \dfrac{[\tau(r,s)] \cdot [\eta(r,s)]^\beta}{\sum_{u \in J_k(r)} [\tau(r,u)] \cdot [\eta(r,u)]^\beta}, & \text{if } s \in J_k(r) \\ 0, & \text{otherwise} \end{cases}$$

(Dorigo et al., 1997)

Each ant also performs a local updating rule in each step so that the next will more likely explore other routes, this process is based on this mathematical formula.

$$\tau(r,s) \leftarrow (1-\rho) \cdot \tau(r,s) + \rho \cdot \Delta\tau(r,s)$$

(Dorigo et al., 1997)

After all the ants have finished building the route, a global updating rule will be performed in which some of the pheromone from all edges evaporates and the best ant adds a pheromone to the route they traveled in proportion to the total distance traveled by the ant. This process is then iterated until it reaches the stopping criteria. The mathematical formula for this step is as follows. (Figure 1)

$$\tau(r,s) \leftarrow (1-\alpha) \cdot \tau(r,s) + \sum_{k=1}^{m} \Delta\tau_k(r,s)$$

(Dorigo et al., 1997)

```
Initialize
Loop
    Each ant is positioned on a starting node
    Loop
        Each ant applies a state transition rule to choose a city
        Each ant applies a local updating rule
    Until all ants have built a complete solution
    A global updating rule is applied
Until End condition
```

Figure-1 Ant Colony System's Pseudocode

4. Results and Discussion

4.1 Mathematical Formula

The mathematical model of this problem is based on the Miller–Tucker–Zemlin TSP mathematical model (Miller et al, 1960). The mathematical model is as follows.

Notation:

N Target Nodes
M Optional Nodes
p Starting Node UAV dan GV
q Ending Node UAV dan GV
A $N \cup M \cup p \cup q$
d_{ij} The distance from i to j
V UAV Velocity (assumed constant)
t_{ij} Duration of UAV from i to j
$t_{ij} = \frac{d_{ij}}{V}$
ST_j Service time of j
T_j Total flying time of UAV when finished serving j
$u_{i,j}$ Number of nodes visited before arriving at i, j
r_j Number of optional nodes visited before arriving at i, j
T_{max} UAV maximum duration

Decision Variable:

$$x_{ij} = \begin{cases} 1, & if\ UAV\ flown\ from\ i\ to\ j \\ 0, & otherwise \end{cases}$$

Objective Function:

$$Min\ Z = \sum_{i=0}^{N} ST_i + \sum_{i=0}^{A} \sum_{j=0}^{A} t_{ij} \cdot x_{ij} \qquad (5.1)$$

Constraints:

$$\sum_{i=0, i \neq j}^{A} x_{ij} = 1, \quad \forall\ j \in N \qquad (5.2)$$
$$\sum_{j=0, j \neq i}^{A} x_{ij} = 1, \quad \forall\ i \in N \qquad (5.3)$$
$$\sum_{i=0, i \neq j}^{A} x_{ij} \leq 1, \quad \forall\ j \in M \qquad (5.4)$$
$$\sum_{j=0, j \neq i}^{A} x_{ij} \leq 1, \quad \forall\ i \in M \qquad (5.5)$$
$$\sum_{i=0}^{A} x_{ip} = 0 \qquad (5.6)$$
$$\sum_{j=0}^{A} x_{pj} = 1 \qquad (5.7)$$
$$\sum_{i=0}^{A} x_{iq} = 1 \qquad (5.8)$$
$$\sum_{j=0}^{A} x_{qj} = 0 \qquad (5.9)$$
$$T_j \leq T_{max}(r_j + 1) \qquad (5.10)$$
$$u_i + 1 \leq u_j + n(1 - x_{ij}) \quad \forall \quad i, j\ \in A, i \neq j \qquad (5.11)$$
$$x_{i,j}\ \in \{0, 1\} \quad \forall \quad i, j\ \in A \qquad (5.12)$$

Equation (5.1) is an objective function to minimize the flight duration of the UAV. Equations (5.2) and (5.3) ensure that the UAV visits and exits node target. This equation also ensures that the UAV only visits and exits once. Equations (5.4) and (5.5) ensure that node optional. Equations (5.6) and (5.7) ensure that node is only for take-off and not visited. Equations (5.8) and (5.9) ensure that nodes are only for landing and not for exiting. Equation (5.10) ensures that the UAV does not run out of energy while carrying out the task. Equation (5.11) ensures that no subtour. Equation (5.12) is a binary constraint.

4.2 Data Collection

The data for this study were obtained from a previous study by Larasati (2019) provided by *Kabupaten Sleman* Regional Disaster Management Agency in which there were 13 nodes target nodes optional. The duration of each edge and service time each node on this network is represented in Table-1 and Table-2. Figure-2 shows the location of each node on a Google Map. Nodes with names in red are nodes optional.

Table-1. Service time of nodes

Node	Nama	Service Time	Node	Nama	Service Time
0	Srunen	3.9	9	Manggang	3.91
1	Kalitengah Kidul	4.44	10	Ngepringan	5.3
2	Kalitengah Lor	5.36	11	Palemsari	4.29
3	Bakalan	4.19	12	Pangukrejo	4.74
4	Kaliadem	4.18	13	Balai Desa Wukirsari	0
5	Jambu	5.19	14	Balai Desa Sindumastani	0
6	Petung	4.78	15	Barak Plosokerep	0
7	Kopeng	4	16	SMP Watuadeg	0
8	Batur	3.5	17	Balai Desa Hargobinangu	0

Table-2. Travel duration for each nodes

Start Node (i)	*End Node (j)*	*Flying Time (min)*	*Start Node (i)*	*End Node (j)*	*Flying Time (min)*
0	1	1.19	9	8	5.45
0	4	0.49	9	10	2.5
1	0	1.19	10	3	2.36
1	2	0.96	10	9	2.5
2	1	0.96	10	16	5.3
3	10	2.36	10	17	2.5
3	13	1.64	11	4	1.08
3	14	5.01	11	5	1.08
4	0	0.49	11	12	0.76
4	5	0.61	12	6	1.49
4	11	1.08	12	11	0.76
5	4	0.61	12	15	2.14
5	6	1.31	13	3	1.64
5	11	1.08	13	14	5.01
6	5	1.31	13	16	1.48
6	7	0.66	14	3	5.01
6	8	0.68	14	13	5.01
6	12	2.54	15	6	2.15
6	15	2.15	15	8	2.35
7	6	0.66	15	12	2.14
7	8	0.38	15	17	2.23
7	9	2.33	16	10	5.3
8	6	0.68	16	13	1.48
8	7	0.38	16	17	2.38
8	9	5.45	17	10	2.5
8	15	2.35	17	15	2.23
9	7	2.33	17	16	2.38

4.3 Results and Discussion

Suppose UAV starts the mission from *SMP Watuadeg* (node 16) and ends at *Balai Desa Hargobinangum* (node 17). The energy limit of the UAV in this case is 60 minutes. This assumption is based on the research of Pólka et al. (2017). After running the program, the solution is represented in Figure-2 with the best results for each iteration represented in Figure-3.

```
In [19]: main()

best path overall=  [[16, 13, 3, 10, 9, 7, 6, 5, 4, 0, 1, 2, 1, 0, 4, 11, 12, 15, 12, 11, 5, 6, 8, 15, 17],
['pass', 'pass', 'serve', 'serve', 'serve', 'serve', 'pass', 'pass', 'serve', 'serve', 'serve', 'serve', 'pas
s', 'pass', 'pass', 'pass', 'pass', 'recharge', 'serve', 'serve', 'serve', 'serve', 'serve', 'pass', 'serve']]
best distance overall=  90.48
```

Figure-2. Results of ACS Program for 2010 Merapi Case

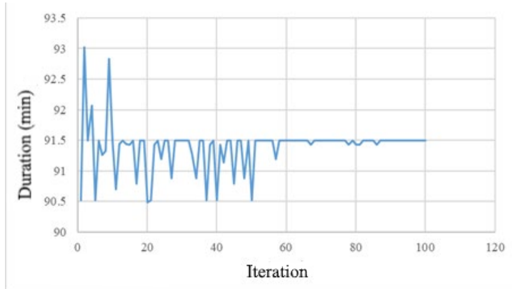

Figure-3. Graph of Best Results for Each Iteration

These results show that the best route found by the algorithm has a time of 90.48 minutes, but the algorithm converges to a route that has a time of 91.5 minutes. This may be due to *Ground Insertion* which makes the shortest route longer at the end. The UAV route with 90.48 minutes takes the following route:

16 – 13P – 3 – 10 – 9 – 7 – 6P – 5P – 4 – 0 – 1 – 2 – 1P – 0P – 4P – 11P – 12P – 15R – 12 – 11 – 5 – 6 – 8 – 15P – 17

The node with the letter **P** indicates that the UAV is only going through *the node* and *the node* with the letter **R** is *the node* where the UAV recharges. From this route, it can be concluded that GV only needs to be at *node* 15, namely the *Barak Plosokerep*. To ensure that this route does not exceed the time limit of the UAV, *log* of the UAV is calculated and represented in Table-3.

Table-3. Duration Log

Node	Action	UAV Duration
16		0
13	pass	1,48
3	serve	3,12
10	serve	5,48
9	serve	12,17
7	serve	19,8
6	pass	24,37
5	pass	25,68
4	serve	26,29
0	serve	31,97
1	serve	37,34
2	serve	42,2
1	pass	47,6
0	pass	48,79
4	pass	49,28
11	pass	50,36
12	pass	51,12
15	recharge	53,26
12	serve	2,14
11	serve	2,9
5	serve	8,72
6	serve	14,32
8	serve	20,19
15	pass	27,32
17	serve	29,55

From this table it can be seen that the UAV did not exceed the time limit, so this route is valid. So, the final UAV route is:

SMP Watuadeg – Balai Desa Wukirsari (P) – Bakalan – Ngepringan – Manggang – Kopeng – Petung (P) – Jambu (P) – Kaliadem – Srunen – Kalitengah Kidul – Kalitengah Lor – Kalitengah Kidul (P) – Srunen (P) – Kaliadem (P) – Palemsari (P) – Pangukrejo (P) – Barak Plosokerep (R) – Pangukrejo – Palemsari – Jambu – Petung – Batur – Barak Plosokerep (P) – Balai Desa Hargobinangum

GV only needs to be at the *Barak Plosokerep* to recharge the UAV's energy. So, the GV route is *SMP Watuadeg – Barak Plosokerep – Balai Desa Hargobinangun*. The final route of the UAV and GV is illustrated in Figure-4.

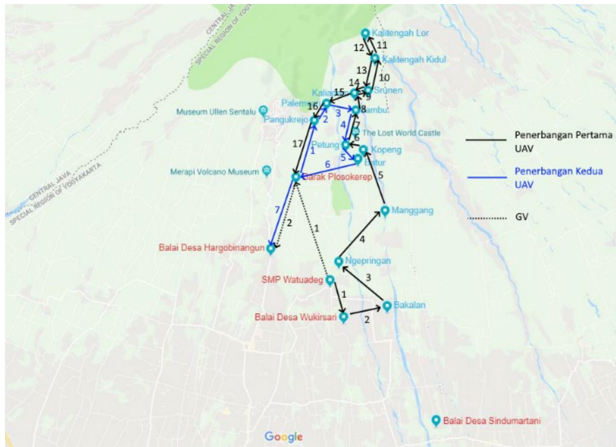

Figure-4 UAV and GV routes in the 2010 Merapi case

Research by Larasati (2019) resulted in a route with a time of 95.48 minutes using the *Depth-First Search* in the 2010 Merapi case. The route taken is as follows: (Figure 5)

SMP Watuadeg – Ngepringan – Bakalan – Ngepringan (P) – Manggang – Batur – Kopeng – Petung – Pangukrejo – Palemsari – Pangukrejo (P) – Barak Plosokerep (R) – Petung (P) – Jambu – Kaliadem – Srunen – Kalitengah Kidul – Kalitengah Lor – Kalitengah Kidul (P) – Srunen (P) – Kaliadem (P) – Jambu (P) – Petung (P) – Barak Plosokerep (P) – Balai Desa Hargobinangun.

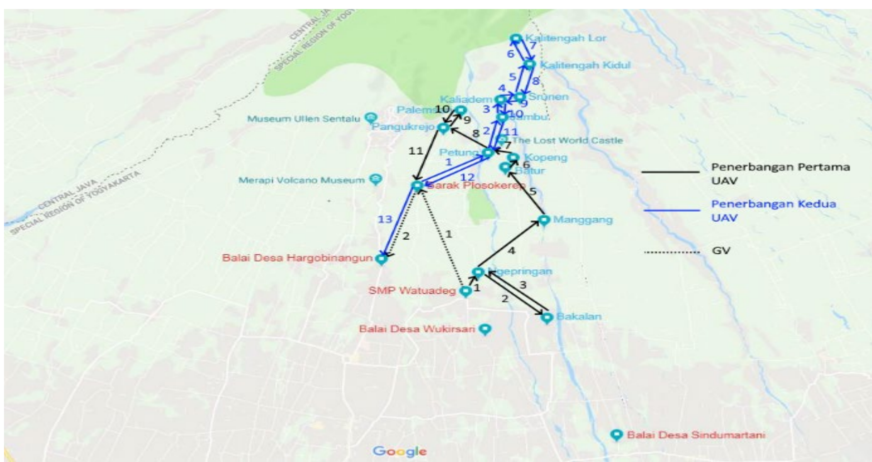

Figure-5. UAV and GV routes by Larasati

These two routes seem to prioritize two different areas, where the route by Larasati serves *the nodes* in the middle first then serves *the nodes* in the north while the ACS route serves *the nodes* in the north and then serves *the nodes* which is in the middle. This difference is probably the reason the route from ACS is faster than the route by Larasati. Another thing that can be found is in serving Bakalan, the Larasati route through Ngepringan first. This is a *time loss* because by going through the Wukirsari Village Hall, you can serve Bakalan more quickly. This can be proven by looking at *log* Larasati's route durationThe following is a *log* of the duration of the route. (Table 4)

Table-4 *Log* Larasati Route Duration

Node	Action	UAV Duration
16		0
10	serve	4,88
3	serve	7,24
10	pass	14,9
9	serve	17,4
8	serve	25,41
7	serve	29,7
6	serve	33,86
12	serve	40,4
11	serve	45,94
12	pass	51,44
15	recharge	53,58
6	pass	2,15
5	serve	3,46
4	serve	8,85
0	serve	14,53
1	serve	19,9
2	serve	24,76
1	pass	30,16
0	pass	31,35
4	pass	31,84
5	pass	32,45
6	pass	33,76
15	pass	35,91
17	serve	38,14

The UAV on the ACS route finished serving Bakalan and Ngepringan in 5.48 minutes while on the Larasati route it finished serving Bakalan and Ngepringan in 7.24 minutes. There is *time loss* of 1.76 minutes. *time loss* may come from the same thing where the Larasati route can't do *a shortcut* but because of the limitations of DFS in determining the route.

5. Conclusion

This research has succeeded in making models and programs for determining UAV routes and coordinating between UAVs and GVs using Ant Colony Optimization (ACO). The type of ACO used is the Ant Colony System which is a development of the Ant System that aims to solve Traveling Salesman Problem (TSP).

This study succeeded in finding the optimal route in the 2010 Merapi Case. This case consisted of 13 nodes target nodes optionalThe routes that have been generated are as follows: *SMP Watuadeg – Balai Desa Wukirsari (P) – Bakalan – Ngepringan – Manggang – Kopeng – Petung (P) – Jambu (P) – Kaliadem – Srunen – Kalitengah Kidul – Kalitengah Lor – Kalitengah Kidul (P) – Srunen (P) – Kaliadem (P) – Palemsari (P) – Pangukrejo (P) – Barak Plosokerep (R) – Pangukrejo – Palemsari – Jambu – Petung – Batur – Barak Plosokerep (P) – Balai Desa Hargobinangum*.

These results indicate that all targets have been visited and from log the duration of the UAV has not exceeded the time limit. GV only visited the Plosokerep Barracks to recharge the UAV's energy. The duration of the route generated by 90.48 minutes; this route is 5 minutes faster than the route generated by the Depth-First Search by Larasati research (2019). This difference is mostly caused by the DFS method which cannot find from one node to shortcuts another, so it takes longer.

References

Asih, A. M. S., Sopha, B. M., Kriptaniadewa, G., Comparison study of metaheuristics: Empirical application of delivery problems. *International Journal of Engineering Business Management*, 9, 1 -12, 2017.

Biggs, N.I., Lloyd, E.K., Wilson, R.J., Graph Theory, *Claredon Press*, 1976.

Chopard, B., Tomassini, M., An Introduction to Metaheuristics for Optimization, Switzerland, *Natural Computing Series, Springer*, 2018.

Dewantoro, R. W., Optimasi Algoritma Ant Colony Optimization (ACO) dengan Menggunakan Algoritma Tabu Search dalam menyelesaikan Traveling Salesman Problem, Medan, Universitas Sumatera Utara, 2019.

Djikstra, E. W., A Note on Two Problems in Connexion with Graphs, *Numerische Mathematik*, S. 269–271, 1959.

Dorigo, M., Birattari, M., Stiitzle, T., Ant Colony Optimization: Artificial Ants as a Computational Intelligence Technique, Belgium, Université Libre de Bruxelles, 2006.

Dorigo, M., Gambardella, L. M., Ant Colony System: A Cooperative Learning Approach to the Traveling Salesman Problem*, IEEE Transactions on Evolutionary Computation*, Vol. 1, No. 1, 1997.

Dorling, K., Heinrichs, J., Messier, G. G., and Magierowski, S., Vehicle routing problems for drone delivery. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 47(1), pp. 70-85, 2017.

Larasati, M.R., Routing Problem for Cooperated Unmanned Aerial Vehicle and Ground Vehicle in Humanitarian Logistics, *UGM Repository*, 2019.

Li, B., Patankar, S., Moridian, B., and Mahmoudian, N., Planning Large-Scale Search and Rescue using Team of UAVs and Charging Stations, *IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*, pp. 1-8, 2018.

Li, Z., and Garcia-Luna-Aceves, J. J., Finding Multi-constrained Feasible Paths by Using Depth-First Search. *Wireless Networks*, 13(3), pp. 323-334, 2006.

Luo, Z., Liu, Z., and Shi, J., A two-echelon cooperated routing problem for a ground vehicle and its carried unmanned aerial vehicle. *Sensors*, 17(5), pp. 1144, 2017.

Luo, Z., Liu, Z., Shi, J., Wang, Q., Zhou, T., and Liu, Y., The Mathematical Modeling of the Two-Echelon Ground Vehicle and Its Mounted Unmanned Aerial Vehicle Cooperated Routing Problem. *IEEE Intelligent Vehicles Symposium (IV)*, pp. 1163-1170, 2018.

Manyam, S. G., Rasmussen, S., Casbeer, D. W., Kalyanam, K., and Manickam, S., Multi-UAV routing for persistent intelligence surveillance and reconnaissance missions, *International Conference on Unmanned Aircraft Systems (ICUAS)*, pp. 573-580, 2017

Mara, S.T.W., Rifai, A. P., Sopha, B. M., An adaptive large neighborhood search heuristic for the flying sidekick traveling salesman problem with multiple drops, *Expert Systems with Applications,* (205),117647, 2022.

Miller, C. E., Tucker, A. W., Zemlin R. A., Integer Programming Formulation of Traveling Salesman Problems, J. ACM 7, 4, 326-329, 1960.

Prins, C., A simple and effective evolutionary algorithm for the vehicle routing problem, *Comput. Oper. Res.*, 31, 1985–2002, 2004.

Prins, C.; Lacomme, P.; Prodhon, C., Order-first split-second methods for vehicle routing problems: A review. *Transp. Res. Part C Emerg. Technol*, 40, 179–200, 2014.

Redi, A.A.N.P., Sopha, B.M., Asih, A.M.S., Liperda, R.I., Collaborative Hybrid Aerial and Ground Vehicle Routing for Post-Disaster Assessment. *Sustainability*, 13, 12841, 2021.

Ross, I. M., Proulx, R. J., Karpenko, M., An Optimal Control Theory for the Traveling Salesman Problem and Its Variants, Naval Postgraduate School, 2020.

Santosa, B., Willy, P., Metoda Metaheuristik: Konsep dan Implementasi, Surabaya, Penerbit Guna Widya, 2011.

Tyas, Y. S, Prijodiprodjo, W., Aplikasi Pencarian Rute Terbaik dengan Metode Ant Colony Optimazation (ACO), Indonesian Journal of Computing and Cybernetics Systems, Vol. 7, No. 1, pp. 55, 2013.

Yakici, E., Solving location and routing problem for UAVs. *Computers and Industrial Engineering*, 102, pp. 294-301, 2016.

Biography

Bertha Maya Sopha is an Associate Professor of Industrial Engineering Program, Department of Mechanical and Industrial Engineering, Universitas Gadjah Mada, Indonesia. She was a former head of the Laboratory of Supply Chain Engineering and Logistics from 2013 to 2015, and a former director of the Industrial Engineering Undergraduate Program from 2016 to 2021. She currently serves as a chair of the Indonesian Association of Industrial Engineering Higher Education Institution (BKSTI), and vice-president of the IEOM Indonesia Chapter. She earned a Bachelor of Engineering (best graduate) from Universitas Gadjah Mada, a master's degree in Management of Production specialization in Transportation and Logistics (graduate with distinction) from the Department of Industrial Economics and Technology Management, Chalmers University of Technology, Sweden. She holds a Ph.D. from the Industrial Ecology Programme, Norwegian University of Science and Technology (NTNU), Norway. She has maintained a high quality of research throughout her academic career including international scholarly leadership in supply chain management and logistics, industrial ecology, and complex system modeling. She has also received various academic achievements, awards, and recognitions such as Distinguished Woman in Industry and Academia

(WIIA) by IEOM Society, Editor Choice Award 2020 by Maritime Economics and Logistics Journal (Palgrave Macmillan), the Best Lecturer runner-up 2015 by Universitas Gadjah Mada, best paper awards at several international and national conferences, and research grantee awards from both Indonesia and abroad institutions. She has professional and community engagement activities to significantly improve the university's reputation through industrial projects and community services.

Alfredo Aryasena graduated from Universitas Gadjah Mada with a bachelor's degree in Industrial Engineering in 2022. He was a lab assistant in Quality and Reliability Engineering Lab and has worked on several projects dan lectures. He is now an Industrial Engineering Post-Graduate Student in Universitas Gadjah Mada. His research interests include optimization, artificial intelligence, machine learning, and simulation.