

# Determination of Optimal Route for Distribution of Rice Food Assistance in City X using the Cuckoo Search Algorithm

Eri Wirdianto<sup>1\*</sup>, Muhamad Gustaffo Bintang<sup>1</sup>, Farhan Azhari<sup>1</sup>

## ABSTRACT

As distribution requirements become more complex, various VRP variants have been developed to address specific challenges. Among them, the Vehicle Routing Problem with Time Windows (VRPTW) adds the constraint that each customer must be served within a predefined time frame, making the routing process considerably more difficult. In this study, we propose a new approach that uses the Cuckoo Search (CS), a metaheuristic algorithm based on Lévy flights. This combination of Lévy flights and CS is designed to maximize exploration of optimal solutions while allowing dynamic route variations. This study aims to evaluate the effectiveness of this approach in optimizing rice food assistance distribution routes in City X. The exact method yielded a global optimum solution consisting of four routes with a minimum total distance of 222 km, requiring 2 hours 36 minutes 27 seconds of computation. In comparison, the cuckoo search algorithm also produced four routes, with a slightly longer total distance of 244 km but a significantly shorter computation time of 1 minute 39 seconds. This represents a difference of about 22 km (9.91%) from the exact solution, demonstrating that for VRPTW, the cuckoo search algorithm can generate near-optimal results with minimal deviation and substantially reduced computational effort.

## Keywords

Route, VRPTW, Exact Method, MILP, Cuckoo Search

<sup>1</sup> Department of Industrial Engineering, Faculty of Engineering, Universitas Andalas  
Limau Manis, Pauh, Padang, West Sumatera

\* Corresponding Author: [eri\\_wirdianto@eng.unand.ac.id](mailto:eri_wirdianto@eng.unand.ac.id)

Submitted : January 30, 2026. Accepted : February 22, 2026. Published : March 02, 2026.

## INTRODUCTION

Efficient distribution of goods is a crucial element of supply chain management, especially in responding to customer demand for fast, low-cost delivery. Shipping route optimization has been a major focus [1]. Optimization is an effort to achieve the best conditions or actions to solve a decision problem with limited existing resources [2]. The vehicle routing problem (VRP) emerged as one of the main problems in logistics research [3], where the goal of VRP is to determine the optimal delivery route to meet the needs of customers in various locations promptly [4][5].

As the complexity of distribution needs increases, various VRP variants have been developed to address more specific challenges. One of them is the capacitated vehicle routing problem (CVRP), which considers vehicle capacity to improve operational efficiency [5][6]. In addition, the vehicle routing problem with time windows (VRPTW), as one of the variants of CVRP, is also a challenge in itself, where VRPTW adds a limitation, each customer has a specific time window, which means the vehicle must arrive at the customer's location within a certain time frame [7].

VRPTW is a crucial problem that directly impacts operational, economic, and environmental efficiency. VRPTW is often encountered in scenarios such as freight forwarding, garbage collection, and healthcare, where meeting service time limits is critical to maintaining customer satisfaction and ensuring regulatory compliance. One example of a delivery scenario in VRPTW is research by Widayani, et al. [8] on the problem of picking up and dropping laundry in the laundry business, where each customer has a specific schedule for receiving or delivering laundry cloth. In addition, in other scenarios such as waste collection, Buhrkal, et al. [9] studied the optimization of waste-collection routes in urban areas under strict time constraints.

Research with VRPTW can also be developed and combined with other constraints, such as that carried out by Kumar, et al. [10] where they solved VRPTW problems using a multi-objective approach. In addition, there is also VRPTW research on the multi-depot case [11], and VRPTW research to minimize the mileage [3].

In solving the VRPTW problem, various methods have been used, ranging from the Adaptive Large Neighborhood Search (ALNS) method to solve the problem of garbage collection in urban areas [9]. ALNS is a metaheuristic method that modifies the initial solution using a destruction-and-repair mechanism. In addition, the Fitness Aggregated Genetic Algorithm (FAGA), which is a combination of fitness in the Genetic Algorithm (GA), has also been used to solve the multi-objective VRPTW problem [10]. Other metaheuristic methods, such as the Ant Colony System (ACS), has also been applied to solve the VRPTW problem with multi-depot cases [11]. The ACS is one variant of the Ant Colony Optimization (ACO) algorithm, which is a metaheuristic algorithm inspired by the behavior of ant colonies in search of food [11]. In addition, there is also a study that was conducted by Alzaqebah, et al. [12] shows how the modified artificial bee colony can outperform the original artificial bee colony in solving the VRPTW problem.

In this study, we propose a new approach using Cuckoo Search (CS), a metaheuristic algorithm based on Lévy flights, which is simpler and more structured than ALNS. This combination of Lévy flights and CS is designed to maximize exploration of optimal solutions while allowing dynamic route variations. This study aims to evaluate the effectiveness of this approach in optimizing rice food assistance distribution routes in City X. It is hoped that the results will contribute to creating a more efficient distribution system in terms of total travel distance, as well as serving as an optimal model for solving VRPTW problems in areas with similar needs.

## METHOD

This study examines a variant of the VRP that focuses on the distribution of rice as a form of food assistance. The rice food assistance program is one of the government's flagship programs to help people experiencing food insecurity. The distribution of rice food assistance that will be studied utilizes 4 vehicles with an average speed of 40 km/h, which will be delivered to 20 points from the Bulog warehouse, where the distribution process is carried out for 11 working hours (07.00 WIB – 18.00 WIB) and fulfilled in a day (one-day service). Each point has opening and closing service hours, so that the VRP problem in this study will be solved with the VRP with Time Windows (VRPTW) approach.

VRP problems can be solved using exact algorithms and approximations. Exact algorithms seek the best solution by thoroughly evaluating all possible solutions, while approximation algorithms aim to find the optimal solution more quickly through estimating near-optimal answers without exploring every option [13]. In this study, exact and approximation algorithms will be used, with the exact algorithm used to assess the extent to which the approximation algorithm's results differ from the optimal solution.

Some of the main types of approximation algorithms are heuristic and metaheuristic. Metaheuristics can be formally defined as iterative processes that guide basic heuristics by intelligently integrating various concepts to explore and utilize the search space. Learning strategies are applied to process information, enabling a more efficient search for optimal solutions [14]. The problem of determining the route of delivering rice food aid in City X in this study was solved using the metaheuristic method, namely the Cuckoo Search Algorithm (CSA) equipped with Lévy flights to find new solutions.

### Exact Method

The exact method explores all possible solutions to find an optimal solution. The exact method is usually used when the problem solved is not too large; in the context of VRP, the problem is not exceeding 150 nodes [15]. The exact method has limitations in dealing with large-scale VRP problems. The computational complexity of this method increases exponentially as the number of nodes increases, making it inefficient for problems with a large number of nodes [16][17]. The exact method in this study will use Mixed Integer Linear Programming (MILP) to get a solution of the VRPTW, which is adapted from Setiawan, et al. [18], who models the problem of Heterogeneous Vehicle Routing Problem with Multi-Trips and Multi-Products (HVRPMTMP). This study will only adapt part of the CVRP model and add it to the VRPTW, with time windows for each customer. The following is a mathematical model of VRPTW.

Sets:

$N$  = Set of nodes (0, 1, 2, ..., N), where node 0 as a depot

$A$  = Set of arcs  $\{(i, j): i, j \in N \text{ and } i \neq j\}$

$k$  = Set of vehicles or routes (1, 2, ..., K)

Parameters:

$Q_i$  = Demand of node  $i$

$CAP_k$  = Capacity of vehicle  $k$

$D_{ij}$  = Distance from customer  $i$  to  $j$

$a_i$  = Earliest service start time at node  $i$

$b_i$  = Latest service start time at node  $i$

$w_{ik}$  = Arrival time of vehicle  $k$  at node  $i$

Objective Function:

$$\text{Min } Z = \sum_{i=0}^N \sum_{j=i+1}^N \sum_{k=1}^K D_{ij} * x_{ijk} \quad (1)$$

Constraints:

$$\sum_{j=1}^N x_{0jk} = 1 \quad \forall k = 1, \dots, K \quad (2)$$

$$\sum_{k=1}^K Y_{ik} = 1 \quad \forall i \in N \quad (3)$$

$$\sum_{j=0, j \neq i}^N x_{ijk} = y_{ij} \quad \forall i \in N, k=1, 2, \dots, K \quad (4)$$

$$\sum_{j=0, j \neq i}^N x_{jik} = y_{ij} \quad \forall i \in N, k=1, 2, \dots, K \quad (5)$$

$$\sum_{i=1}^N Q_i Y_{ik} \leq CAP_k \quad \forall k=1, 2, \dots, K \quad (6)$$

$$\sum_{j=1}^N x_{j0k} = 1 \quad \forall k = 1, 2, \dots, K \quad (7)$$

$$w_{ik} + t_{ik} + s_i - w_{jk} \leq M*(1 - x_{ijk}) \quad \forall i \in N, j \in N, k=1, 2, \dots, K \quad (8)$$

$$a_i \leq w_{ik} \leq b_i \quad \forall i \in N, k=1, 2, \dots, K \quad (9)$$

$$u_i - u_j + N x_{ijk} \leq N - 1 \quad \forall i \in N, j \in N, k = 1, 2, \dots, K, \quad (10)$$

$$u(1) = 0 \quad (11)$$

$$y_{ik} \in \{0, 1\} \quad \forall i \in N, k = 1, 2, \dots, K \quad (12)$$

$$x_{ijk} \in \{0, 1\} \quad \forall i \in N, j \in N, k = 1, 2, \dots, K \quad (13)$$

Based on the mathematical model above, the search for the optimal solution is carried out using Lingo 21.0 software, which employs the Branch-and-Bound algorithm. The results obtained using the algorithm are the global optimum of a problem.

### Cuckoo Search Algorithm

The cuckoo search algorithm is a metaheuristic method introduced by Yang and Deb in 2009, inspired by the obligate parasitic behavior in several species of cuckoos [19][20]. The CSA algorithm is based on three simple rules or the following assumptions:

1. Each cuckoo bird lays only one egg and places it in a randomly selected nest.
2. The best eggs with high quality will survive or be passed on to the next generation.
3. The number of host nests is fixed. Therefore, the probability that the host identifies the cuckoo egg is given by the probability that  $p_a \in [0,1]$ . If the host recognizes the egg, it can throw it out or leave the nest and build a new one.

Basically, the CSA consists of three main components: selecting the best solution, exploiting local random walks, and exploring using global randomization with a Lévy flights pattern. Lévy flights are a pattern of motion used by many animals and insects to explore their environment, involving a combination of straight paths and sharp changes of direction up to 90 degrees. Lévy flights essentially provide a "random walk" in which the length of each step is drawn from a Lévy distribution.

$$\text{Lévy} \sim u = t - \lambda, (1 < \lambda \leq 3) \quad (14)$$

To set the step size (random pattern) when generating a solution using the Lévy flights, the user-determined  $\alpha$  (alpha coefficient) is used as a controller. In most cases,  $\alpha$  can be defined as 1. When the Lévy step is generated with a random number generator, the result is first multiplied by  $\alpha$  before being used to create a new solution. Cuckoo eggs produced based on the new solution are equivalent to the process of creating a new solution for cuckoo flocks. Figure 1 shows a brief overview of the CSA with Lévy flights logic [19][21].

---

#### ***Cuckoo Search Algorithm***

Objective function  $f(x)$ ,  $x = (x_1, \dots, x_n)^T$

Initiate a random population of  $n$  host nests,  $x_i$  ( $i = 1, 2, \dots, n$ )

**While** ( $t < \text{Max generation}$ ) or (Stop criteria)

Get a cuckoo randomly using Lévy flight

Evaluate its quality or fitness  $F_i$  of that cuckoo

**If** ( $F_i > F_j$ ),

Replace  $j$  with that new solution;

**End if**

A fraction ( $p_a$ ) of worse nests is abandoned and new ones are built;

Keep the best solutions (or nests with quality solutions);

Rank the solutions and find the current best

**End while**

Postprocess results and visualization

**End**

---

Figure 1. Cuckoo search algorithm logic

The following is a description of each symbol used:

- $x_i$  = Position vector or solution to  $i$  in a population.
- $f(x)$  = The objective function evaluated at position  $x$ ; used to assess the quality of the solution.
- $t$  = Index of iterations or generations in the algorithm process.
- $\alpha$  = Scale parameter that controls step size in Lévy flights.
- $pa$  = Probability that the nest will be abandoned or replaced; reflects the level of exploration of the algorithm.
- $n$  = The total number of nests or solutions in the population.
- $\otimes$  = An operator that indicates the multiplication of elements element-wise.

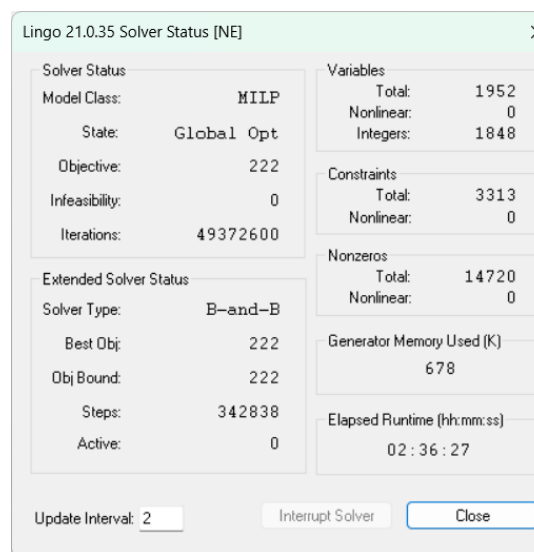
## RESULT AND DISCUSSION

This research uses a case study of rice food assistance distribution in City X. The data used such as distance matrix, demand, time windows for each customer, and travel time and service time can be seen at the following link: [https://docs.google.com/spreadsheets/d/1ekCz50hrf8MEhIWHhLPTe8He8Y0u\\_US3/edit?usp=sharing&oid=105212444710782226318&rtpof=true&sd=true](https://docs.google.com/spreadsheets/d/1ekCz50hrf8MEhIWHhLPTe8He8Y0u_US3/edit?usp=sharing&oid=105212444710782226318&rtpof=true&sd=true)

Based on the data, this case study was solved using two methods: the exact method in Lingo 21.0 and the CSA in Python 3.0.

### Exact Method Results

The exact method in this study found an optimal solution in 2 hours, 36 minutes, and 27 seconds, with a minimum total distance of 222 km, using 49 million iterations. Details on the solution's status are shown in [Figure 2](#).



[Figure 2](#). Lingo 21.0 solver status for the exact method

The detailed routes provided by Lingo are shown in [Figure 3](#), where  $X_{ijk}$  is selected; index  $i$  indicates the origin node, index  $j$  indicates the destination node, and index  $k$  indicates the vehicle. Node 1 is the depot, and nodes 2-21 are the customers. Visualization of the optimal routes obtained by the exact method can be seen in [Figure 4](#).

```

Global optimal solution found.
Objective value:                222.0000
Objective bound:                222.0000
Infeasibilities:                0.000000
Extended solver steps:         342233
Total solver iterations:       49372600
Elapsed runtime seconds:       9387.75

Variable      Value      Reduced Cost
X( 1, 3, 3)   1.000000   8.000000
X( 1, 5, 2)   1.000000   8.000000
X( 1, 8, 4)   1.000000   8.000000
X( 1, 13, 1)  1.000000  11.000000
X( 2, 1, 4)   1.000000   9.000000
X( 3, 17, 3)  1.000000   9.000000
X( 4, 19, 2)  1.000000   7.000000
X( 5, 4, 2)   1.000000   7.000000
X( 6, 16, 1)  1.000000  10.000000
X( 7, 1, 2)   1.000000  11.000000
X( 8, 12, 4)  1.000000  10.000000
X( 9, 1, 1)   1.000000  13.000000
X( 10, 9, 1)  1.000000   8.000000
X( 11, 1, 3)  1.000000   7.000000
X( 12, 20, 4) 1.000000  12.000000
X( 13, 6, 1)  1.000000   7.000000
X( 14, 11, 3) 1.000000  12.000000
X( 15, 14, 3) 1.000000  11.000000
X( 16, 10, 1) 1.000000   9.000000
X( 17, 15, 3) 1.000000  12.000000
X( 18, 21, 2) 1.000000  10.000000
X( 19, 18, 2) 1.000000   8.000000
X( 20, 2, 4)  1.000000   7.000000
X( 21, 7, 2)  1.000000   8.000000
    
```

Figure 3. Exact method computation results using Lingo 21.0

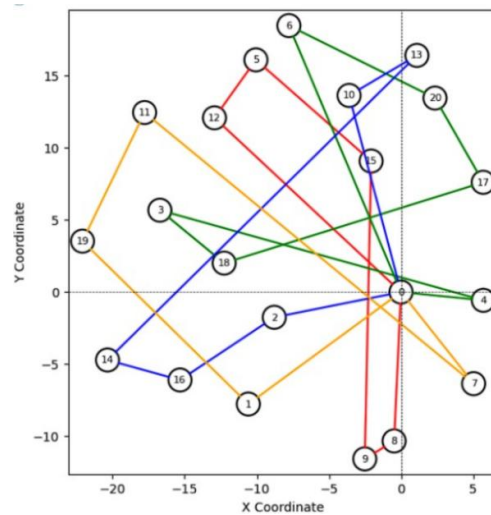


Figure 4. Visualization of the optimal route of the exact method

### Cuckoo Search Algorithm Results

The metaheuristic CSA solved the problem within 1 minute 39 seconds and obtained a minimum total distance of 244 km. The parameters used when running the algorithm are: number of nests = 10,  $p_a = 0.1$ , and iterations = 1000. The CSA parameters were selected based on a combination of literature precedent [21] and preliminary trial runs. The detailed routes generated by CSA using Python are shown in Figure 5.

```

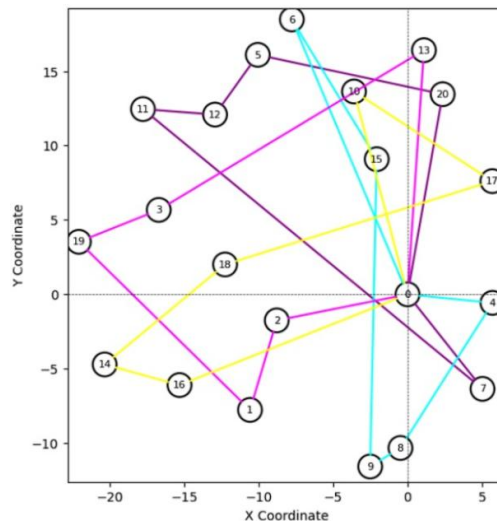
Elapsed Time: 98.94s, Best Distance: 244

Best Solution:
Vehicle 1 = [0, 7, 11, 12, 5, 20, 0]
Distance = 54
Vehicle 2 = [0, 4, 8, 9, 15, 6, 0]
Distance = 55
Vehicle 3 = [0, 2, 1, 19, 3, 13, 0]
Distance = 70
Vehicle 4 = [0, 16, 14, 18, 17, 10, 0]
Distance = 65

Total Best Distance: 244
Total Computation Time: 98.9437 seconds
    
```

Figure 5. Cuckoo search algorithm results using Python 3.0

Visualization of the optimal routes obtained by the exact method can be seen in [Figure 6](#).



**Figure 6.** Visualization of the optimal route of the cuckoo search algorithm

**Comparison of Solution Results**

The exact method, MILP, using the Lingo 21.0 software, and the metaheuristic method with the CSA using Python 3.0 are compared based on the minimum total mileage and computational time to assess the performance of the CSA to get a solution, based on how far the solution is different from the optimal solution in the exact method. The following is a comparison of the two methods, shown in [Tables 1](#) and [2](#).

*Table 1. Comparison of formed route solutions*

Vehicle	Exact Method	Cuckoo Search Algorithm
1	Depot-12-5-15-9-8-Depot	Depot-7-11-12-5-20-Depot
2	Depot-4-3-18-17-20-6-Depot	Depot-4-8-9-15-6-Depot
3	Depot-2-16-14-13-10-Depot	Depot-2-1-19-3-13-Depot
4	Depot-7-11-19-1-Depot	Depot-16-14-18-17-10-Depot

*Table 2. Mileage comparison*

Vehicle	Exact Method	Cuckoo Search Algorithm	Difference
1	58	54	
2	59	55	
3	59	70	9.91%
4	46	65	
Total	222	244	

[Table 2](#) shows that the exact method provides an optimal solution with a minimum total distance of 222 km and a computational time of 2 hours 36 minutes 27 seconds, while the CSA produces a near-optimal solution with a minimum total distance of 244 km and a computational time of 1 minute 39 seconds. The difference in total mileage generated is about 22 km (9.91%). This difference shows that the CSA can provide a near-optimal solution 250 times faster than the exact solution.

## CONCLUSION

This study has evaluated the effectiveness of the proposed approach in optimizing rice food assistance distribution routes within City X, situating the problem within the broader context of humanitarian logistics. Beyond identifying efficient routes for the case study, the research also examined the capability of the cuckoo search algorithm to approximate global optimal solutions for the VRPTW problem.

The exact method yielded a global optimum of four routes with a minimum total travel distance of 222 km, requiring 2 hours 36 minutes 27 seconds of computation. In comparison, the cuckoo search algorithm produced four routes with a total travel distance of 244 km in only 1 minute 39 seconds, reflecting a difference of 22 km (9.91%) from the global optimum. These findings demonstrate that the cuckoo search algorithm can generate solutions that are acceptably close to the global optimum while drastically reducing computational time, underscoring its practical value for large-scale humanitarian logistics applications where efficiency and timeliness are critical. Thus, the cuckoo search algorithm offers a promising balance between solution quality and computational efficiency, making it a valuable tool for humanitarian logistics routing problems where rapid decision-making is essential.

## REFERENCES

- [1] S. Chopra and P. Meindl, *Supply chain management: strategy, planning, and operation*. New York: Pearson, 2013.
- [2] D. Herdhiansyah, A. Yunissa, W. Gusmiarty, and A. Asriani, "Optimasi penentuan rute distribusi beras komersial Bulog menggunakan metode branch and bound dengan bantuan software lingo," *AGROINTEK*, vol. 17, no. 2, pp. 353-364, 2023.
- [3] J. Nalepa and M. Blocho, "Adaptive memetic algorithm for minimizing distance in the vehicle routing problem with time windows," *Soft Computing*, vol. 20, pp. 2309–2327, 2016.
- [4] I. Lukman, R. Hanafi, and S. M. Parenreng, "Optimasi biaya distribusi pada HFVRP menggunakan algoritma particle swarm optimization," *JOSI: Jurnal Optimasi Sistem Industri*, vol. 18, no. 2, pp. 164–175, 2019.
- [5] P. Toth and D. Vigo, *Vehicle routing: problems, methods, and applications*, Second ed. Philadelphia: Society for Industrial and Applied Mathematics (SIAM), 2015.
- [6] R. A. T. Saputro, Y. U. Kasanah, O. R. Marddani, and K. Niami, "Optimasi rute distribusi unggas berbasis network analysis-GIS menggunakan capacitated vehicle routing problem with time window pickup and delivery," *Jurnal INTECH Teknik Industri Universitas Serang Raya*, vol. 10, no. 1, pp. 51–60, 2024.
- [7] T. D. C. Le, D. D. Nguyen, J. Oláh, and M. Pakurár, "Clustering algorithm for a vehicle routing problem with time windows," *TRANSPORT*, vol. 37, no. 1, pp. 17–27, 2022.
- [8] W. Widayani, D. A. Pratama, R. D. Pratama, E. T. Kusumajaya, and A. Dharma, "Pemanfaatan metode heuristik travelling salesman problem with time windows pada rute antar jemput laundry dengan algoritma genetika," *Jurnal Informatika Komputer, Bisnis, dan Manajemen*, vol. 17, no. 1, pp. 1-10, 2019.
- [9] K. Buhrrkal, A. Larsen, and S. Ropke, "The waste collection vehicle routing problem with time windows in a city logistics context," *Procedia Social and Behavioral Sciences*, vol. 39, pp. 241-254, 2012.
- [10] V. S. Kumar, M. R. Thansekhar, R. Saravanan, and S. M. J. Amali, "Solving multi-objective vehicle routing problem with time windows by FAGA," *Procedia Engineering*, vol. 97, pp. 2176-2185, 2014.
- [11] M. Dubillard, X. Lorca, and M. Lauras, "An ant colony system for the skilled, multi-depot VRP with due dates and time windows," *IFAC-PapersOnLine*, vol. 56, no. 2, pp. 11129-11134, 2023.
- [12] M. Alzaqebah, S. Abdullah, and S. Jawarneh, "Modified artificial bee colony for the vehicle routing problems with time windows," *SpringerPlus*, vol. 5, no. 1298, pp. 1-14, 2016.

- 
- [13] D. Muriyatmoko, A. Djunaidy, and A. Muklason, "Heuristics and metaheuristics for solving capacitated vehicle routing problem: an algorithm comparison," *Procedia Computer Science*, vol. 234, pp. 494-501, 2024.
- [14] M. Iqbal, M. Zarlis, T. Tulus, and H. Mawengkang, "Model pendekatan metaheuristik dalam penyelesaian optimisasi kombinatorial," in *Seminar Nasional Teknologi Komputer & Sains (SAINTEKS)*, Medan, 2020, pp. 92-97: STMIK Budi Darma.
- [15] A. F. R. Silva, A. T. Pozos, D. A. Martinez, and J. W. Escobar, "An iterated local search based approach for a real rich vehicle routing problem with time windows," *Computers & Industrial Engineering*, vol. 208, pp. 1-15, 2025.
- [16] E. Wirdianto, D. Regenie, and Wisnel, "Aplikasi algoritma hybrid dalam penentuan rute pendistribusian produk (studi kasus: PT. Enseval Putera Megatrading)," *Jurnal Optimasi Sistem Industri*, vol. 15, no. 2, pp. 171-180, 2016.
- [17] I. H. Muna, "Performansi analisis algoritma koloni semut (ant colony optimization) dalam menyelesaikan permasalahan capacitated vehicle routing problem (CVRP)," *SCIENCE TECH: Jurnal Ilmu Pengetahuan dan Teknologi*, vol. 8, no. 2, pp. 98-112, 2022.
- [18] F. Setiawan, N. A. Masruroh, and Z. I. Pramuditha, "On modelling and solving heterogeneous vehicle routing problem with multi-trips and multi-products," *Jurnal Teknik Industri*, vol. 21, no. 2, pp. 91-104, 2019.
- [19] A. Nurhayati, "Perbandingan algoritma cuckoo search krill herd (CSKH) dan levy-flight krill herd (LKH)," in *Seminar Nasional Telekomunikasi dan Informatika (SELISIK)* Bandung, 2018, pp. 448-453: Sekolah Tinggi Teknologi Bandung.
- [20] J. H. Santillan, S. Tapucar, C. Manliguez, and V. Calag, "Cuckoo search via Lévy flights for the capacitated vehicle routing problem," *Journal of Industrial Engineering International*, vol. 14, pp. 293-304, 2018.
- [21] S. Shiralkar, A. Bahulekar, and S. Jawade, "The cuckoo search algorithm: a review," *International Research Journal of Engineering and Technology (IRJET)*, vol. 9, no. 8, pp. 1238-1246, 2022.

This page is intentionally left blank