

Ant Colony Optimization for Waste Collection Routing: A Case Study in Sekar Tunjung Residential

Ida Bagus Kade Puja Arimbawa K¹

¹Fakultas Teknologi dan Ilmu Kesehatan, Program Studi Sistem Informasi, Universitas Bali Dwipa, Indonesia

¹kemenuh.puja@gmail.com



*Corresponding Author

Article History:

Submitted: 22-05-2025

Accepted: 29-05-2025

Published: 10-06-2025

Keywords:

Ant Colony Optimization; Waste Collection; Metaheuristic; Shortest Route; Sekar Tunjung Residential.

Brilliance: Research of

Artificial Intelligence is licensed under a Creative Commons Attribution-NonCommercial 4.0 International (CC BY-NC 4.0).

ABSTRACT

Waste collection is a critical issue in the management of efficient urban systems, particularly in densely populated residential areas. Inefficient waste transportation routes can lead to excessive fuel consumption, prolonged collection times, and increased operational costs. This study proposes the application of the Ant Colony Optimization (ACO) algorithm as a metaheuristic approach to solve the Capacitated Vehicle Routing Problem (CVRP) in the context of waste collection. ACO mimics the foraging behavior of ant colonies, allowing it to explore and reinforce optimal routing paths based on pheromone intensity and distance-based desirability. The methodology involves modeling a spatial network of waste pick-up points, each with varying waste volumes, and incorporating truck capacity constraints into the routing algorithm. The case study is situated in the Sekar Tunjung residential in East Denpasar, where the simulation uses ten pick-up points and a waste truck starting and ending at TPS3R Kesiman. Parameters such as pheromone influence (α), visibility (β), and evaporation rate (ρ) are tuned to find the best route configuration. Simulation results show that ACO efficiently constructs a single-trip route covering all points while respecting capacity limits. The algorithm demonstrates adaptability to changes in volume distribution and capacity scenarios, resulting in minimal total travel distance. This research confirms the potential of ACO as a robust and flexible solution for optimizing waste transportation logistics in urban environments.

INTRODUCTION

Sekar Tunjung Residential Area is located in East Denpasar, Bali, and is one of the fastest-growing residential neighborhoods (Badan Pusat Statistik Kota Denpasar, 2024). The area is known for its good accessibility and its proximity to various public facilities such as shopping centers, schools, and healthcare services (99.co, 2024). The rapid development of housing in this area has led to an increase in population and domestic activities, which in turn contributes to the growing volume of household waste. However, the waste collection system in Sekar Tunjung faces several challenges.



Fig. 1 Map of Sekar Tunjung Residential Area

One of the key issues is the inefficiency of waste collection routes, which results in longer travel times and higher fuel consumption. Additionally, the variation in waste volume at each collection point is often not aligned with the capacity of the transport trucks, leading to certain areas being underserved. This problem is further exacerbated by the lack of accurate data regarding waste generation and composition at the local level, which is essential for planning efficient waste collection routes (LCDI Indonesia, 2024).



Waste management remains one of the major challenges in urban logistics systems. An important aspect of this management lies in the efficiency of routing from various collection points to the final disposal site. This process not only demands time and cost efficiency but must also consider vehicle capacity and the volume of waste at each collection point.

Metaheuristic algorithms such as Ant Colony Optimization (ACO) have proven effective in solving routing optimization problems, such as the Travelling Salesman Problem (TSP) and the Capacitated Vehicle Routing Problem (CVRP) (Hu et al., 2024; Ky Phuc & Phuong Thao, 2021). In this context, ACO can be applied to generate optimal waste collection routes based on available spatial and operational data.

This study aims to apply the ACO algorithm to the problem of waste transportation by considering varying waste volumes at each point and the limited capacity of the collection trucks. A case study was conducted in the Sekar Tunjung residential area, Denpasar, with the route starting and ending at TPS3R Kesiman via the main entrance on Tunjung Kuning Street. The simulation was carried out using a sample of 10 collection points, and the results were analyzed based on total route distance and the number of trips required.

LITERATURE REVIEW

Ant Colony Optimization (ACO) was introduced by Dorigo, Maniezzo, and Colorni in 1996 as a metaheuristic inspired by the foraging behavior of ant colonies in discovering the shortest paths to food sources. Ants use pheromone trails to communicate and reinforce the most efficient paths, a mechanism that has been modeled computationally to solve complex combinatorial problems. ACO has since been formalized and further developed as a robust optimization framework (Dorigo & Di Caro, 1999), and has been successfully applied to classic problems such as the Travelling Salesman Problem (TSP), the Vehicle Routing Problem (VRP), and its extensions like the Capacitated Vehicle Routing Problem (CVRP).

In the context of waste collection, numerous studies have demonstrated the effectiveness of ACO and its variants. Hu et al. (2024) developed a hybrid algorithm called the Improved Ant Colony–Shuffled Frog Leaping Algorithm (IAC-SFLA) for domestic waste collection, which successfully reduced travel distance and improved load efficiency. Rattanawai et al. (2024) applied ACO to optimize the collection of both medical and non-medical waste by considering service priorities and time constraints.

Okrah et al. (2023) adapted ACO for waste collection in Ghana by incorporating carbon emissions and social cost considerations, demonstrating high efficiency. Shen et al. (2023) compared ACO with Particle Swarm Optimization (PSO) in a multi-trip urban waste collection context, finding that ACO remained competitive. Algethami (2023) showed that ACO could deliver optimal solutions with efficient computation time when combined with appropriate initialization strategies.

Gunawan et al. (2024) successfully implemented ACO in the waste transportation system of Tegal City, generating adaptive routes based on local spatial data. Manoharan et al. (2025) integrated ACO with an IoT-based Swin Transformer model for smart solid waste management. Wouda et al. (2024) combined ACO with data-driven decision-making to support waste selection and routing in urban areas.

The fundamental concept and applications of ACO were comprehensively formulated by Dorigo and Di Caro (1999), providing the foundation for subsequent developments of ACO variants. These studies reaffirm that ACO remains relevant and flexible for local-scale waste transportation scenarios, especially those with limited vehicle capacity and variable waste volumes, as addressed in this study.

METHOD

This study employs the Ant Colony Optimization (ACO) algorithm to design efficient waste collection routes. Mathematically, the probability of an ant moving from one node to another is calculated using the following formula:

$$P_{ij}^k = \frac{[\tau_{ij}]^\alpha \cdot [\eta_{ij}]^\beta}{\sum_{l \in N_i^k} [\tau_{il}]^\alpha \cdot [\eta_{il}]^\beta} \quad (1)$$

Where:

- τ_{ij} is the pheromone intensity on edge (i, j)
- $\eta_{ij} = \frac{1}{d_{ij}}$ is the heuristic value (desirability) of edge (i, j) , with d_{ij} representing the distance between vertex
- α is the parameter that controls the influence of pheromone
- β is the parameter that controls the influence of desirability
- N_i^k is the set of vertex that ant k from i

After all ants have completed the construction of their solutions, the pheromone levels are updated using the following formula:

$$\tau_{ij} \leftarrow (1 - \rho) \cdot \tau_{ij} + \Delta \tau_{ij} \quad (2)$$

Where:



- $\rho \in (0,1)$ is the pheromone evaporation rate
- $\Delta\tau_{ij}$ is the amount of pheromone added to edge (i,j) and is calculated based on the contributions of the ants as follows:

$$\Delta\tau_{ij} = \sum_{k=1}^m \Delta\tau_{ij}^k \text{ with } \Delta\tau_{ij}^k \begin{cases} \frac{Q}{L_k}, & \text{if ant travels through edge } (i,j) \\ 0, & \text{otherwise.} \end{cases} \quad (3)$$

Where Q is a constant, and L_k is the total length of the route taken by ant k . The shorter the route taken by an ant, the greater the amount of pheromone deposited, thereby encouraging other ants to explore that path.

This study utilizes spatial and operational data to construct a waste collection simulation for the Sekar Tunjung residential area. The spatial data, in the form of distances between waste collection points, were obtained through field mapping and structured into a weighted adjacency matrix using a two-dimensional array (Arimbawa et al., 2023). Points that are not directly connected were assigned an infinite value. A total of 10 collection points were used in the simulation, with the route starting from TPS3R Kesiman as the initial point (vertex v_1) entering through the main gate on Tunjung Kuning Street.

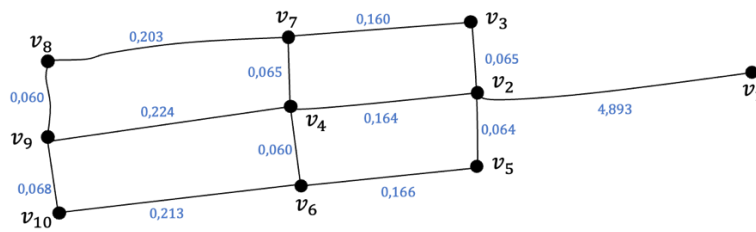


Fig. 2 Graph

This implicitly forms a weighted sparse graph, in which only a small subset of all possible connections between vertex is available in reality. Therefore, this study assumes that a vertex may be traversed more than once, as long as it allows for the determination of a minimum-length route. As an illustration, $d(v_3, v_5) = d(v_3, v_2) + d(v_2, v_5)$. Adjacency Matrix denoted by $A = [a_{ij}]$, is an $n \times n$ matrix defined as follows (Arimbawa K., 2024) :

$$a_{ij} = \begin{cases} 1, & \text{if vertex } v_i \text{ adjacent to vertex } v_j \\ 0, & \text{otherwise} \end{cases} \quad (4)$$

In Fig. 2, the corresponding adjacency matrix is constructed as follows.

	V1	V2	V3	V4	V5	V6	V7	V8	V9	V10
V1	0	4893	INF	INF	INF	INF	INF	INF	INF	INF
V2	4893	0	65	164	64	230	225	428	388	443
V3	INF	65	0	225	129	295	160	363	423	508
V4	INF	164	225	0	226	60	65	284	224	292
V5	INF	64	129	226	0	166	291	507	450	379
V6	INF	230	295	60	166	0	125	328	281	213
V7	INF	225	160	65	291	125	0	203	263	338
V8	INF	428	363	284	507	328	203	0	60	128
V9	INF	388	423	224	450	281	263	60	0	68
V10	INF	443	508	292	379	213	338	128	68	0

Fig. 3 Adjacency matrix

Subsequently, the implementation of the ACO algorithm was carried out using the C++ programming language, which was chosen for its efficiency in handling large-scale iterations and mathematical processing. The entire process—from parameter initialization, solution construction by ants, computation of transition probabilities, to pheromone update—was systematically executed in C++. Data structures such as the distance matrix, visit status of nodes, and waste volume were organized using two-dimensional arrays and vectors. Each iteration was run by 10 ants, repeated for a total of 10 iterations, and the best solution from each iteration was stored for further analysis.

Route construction by the ants considers two main constraints: the maximum truck capacity of 12 m³ and the simulated average waste volume per collection point, which ranges between 0.8 m³ and 1 m³. This means that the truck can still accommodate the total volume as long as it does not exceed its capacity.

RESULT

The Ant Colony Optimization (ACO) algorithm was simulated to design an efficient waste collection route in the Sekar Tunjung residential area. In this simulation, 10 waste collection points were used, with the starting point located at TPS3R Kesiman. The number of ants was set to 10, with a maximum of 10 iterations. The pheromone influence

parameter was set to $\alpha = 1,0$ and the visibility parameter was set to $\beta = 5,0$ o balance between exploration based on past experience (pheromone) and preference for shorter distances (visibility). A higher β value indicates that the distance between nodes significantly influences route selection.

The pheromone evaporation rate was set at $\rho = 0,5$ to prevent the ants from being overly biased toward early solutions and to allow the discovery of new paths. A pheromone update constant of $Q = 100$ was used to reinforce the best-found paths. Initial pheromone values were uniformly distributed, while the visibility between nodes was calculated as $\eta_{ij} = \frac{1}{d_{ij}}$, except when d_{ij} is very large (e.g., INF), n which case the visibility value is considered zero.

The ACO algorithm in this study was implemented using the C++ programming language due to its ability to efficiently handle iterative computation and multidimensional array processing. The code structure is divided into several key components that collectively support the construction of ACO solutions:

1. Graph representation using an adjacency matrix: The distance between waste collection points is modeled as a two-dimensional matrix `vector<vector<int>> D`, where the value `D[i][j]` represents the distance from point i to point j . The value `Nilai INF` s used to indicate that two points are not directly connected (a representation of a sparse graph). This approach aligns with real-world conditions in which direct access between all points is not feasible.
2. ithm parameters: Parameters such as the number of ants (`nAnt`), number of iterations (`nIter`), pheromone influence (`alpha`), visibility (`beta_param`), and evaporation rate (`rho`) are defined as global constants. These parameters control how ants construct solutions and how quickly the algorithm converges toward an optimal solution.
Solution construction by ants: Each ant is represented by a loop that builds a solution by selecting the next node based on a probability:
`prob[j] = pow(tau[current][j], alpha) * pow(eta[current][j], beta_param);`
3. Truck capacity control: During the node selection process, only nodes with waste volumes that do not exceed the remaining truck capacity are eligible to be visited. This logic is enforced by the condition `waste[j] <= cap` ensuring that the generated route is operationally feasible. In this case, the truck capacity is set at $12m^3$.
4. Pheromone update: After all ants have completed their routes, the pheromone levels are updated based on the quality of the generated routes using the following formula:
`tau[from][to] += Q / len;`
`tau[to][from] += Q / len;`
Pheromone trails are also globally evaporated to avoid premature convergence:
`tau[i][j] *= (1 - rho);`
5. Evaluation and storage of the best solution: The total route distance is recorded and compared in each iteration. If a shorter route is found, that solution is recorded as the best solution throughout the iterations:
`if (len < bestLength) {`
 `bestLength = len;`
 `bestRoute = route;`
`}`

During the simulation process, the algorithm was executed over 10 iterations with 10 ants in each iteration. Each ant incrementally constructed a route while considering two main constraints: the maximum capacity of the waste collection truck and the average waste volume per collection point. If the truck reached its capacity, the ant was directed back to the depot to begin a new sub-route. However, under the current configuration, all collection points could be served in a single trip without requiring additional sub-routes. The following presents the simulation results from each iteration:

Table 1. Ant Routes in Iteration 1

Ant	Routes (v)	Distance(meters)
Ant 1	1 -> 2 -> 3 -> 7 -> 4 -> 6 -> 5 -> 9 -> 10 -> 8 -> 1	6055
Ant 2	1 -> 2 -> 3 -> 5 -> 6 -> 4 -> 7 -> 9 -> 8 -> 10 -> 1	5829
Ant 3	1 -> 2 -> 5 -> 3 -> 7 -> 4 -> 6 -> 9 -> 8 -> 10 -> 1	5840
Ant 4	1 -> 2 -> 6 -> 4 -> 7 -> 8 -> 9 -> 10 -> 5 -> 3 -> 1	6087
Ant 5	1 -> 2 -> 5 -> 3 -> 7 -> 4 -> 6 -> 10 -> 9 -> 8 -> 1	5712
Ant 6	1 -> 2 -> 5 -> 6 -> 4 -> 7 -> 3 -> 8 -> 9 -> 10 -> 1	5899
Ant 7	1 -> 2 -> 3 -> 5 -> 4 -> 6 -> 7 -> 8 -> 9 -> 10 -> 1	5829
Ant 8	1 -> 2 -> 3 -> 5 -> 6 -> 4 -> 7 -> 8 -> 9 -> 10 -> 1	5709
Ant 9	1 -> 2 -> 5 -> 6 -> 4 -> 7 -> 8 -> 9 -> 10 -> 3 -> 1	6087
Ant 10	1 -> 2 -> 3 -> 7 -> 4 -> 6 -> 5 -> 10 -> 9 -> 8 -> 1	5916

Table 2. Ant Routes in Iteration 2

Ant	Routes (v)	Distance(meters)
Ant 1	1 -> 2 -> 5 -> 3 -> 7 -> 4 -> 6 -> 10 -> 9 -> 8 -> 1	5712
Ant 2	1 -> 2 -> 5 -> 3 -> 7 -> 4 -> 6 -> 10 -> 9 -> 8 -> 1	5712
Ant 3	1 -> 2 -> 3 -> 7 -> 4 -> 6 -> 5 -> 9 -> 8 -> 10 -> 1	6047
Ant 4	1 -> 2 -> 3 -> 7 -> 4 -> 6 -> 5 -> 10 -> 9 -> 8 -> 1	5916
Ant 5	1 -> 2 -> 3 -> 7 -> 4 -> 6 -> 5 -> 10 -> 9 -> 8 -> 1	5916
Ant 6	1 -> 2 -> 3 -> 7 -> 4 -> 6 -> 5 -> 10 -> 9 -> 8 -> 1	5916
Ant 7	1 -> 2 -> 3 -> 7 -> 4 -> 6 -> 10 -> 9 -> 8 -> 5 -> 1	6091
Ant 8	1 -> 2 -> 5 -> 3 -> 4 -> 6 -> 7 -> 9 -> 8 -> 10 -> 1	5947
Ant 9	1 -> 2 -> 3 -> 10 -> 9 -> 8 -> 4 -> 7 -> 6 -> 5 -> 1	6234
Ant 10	1 -> 2 -> 3 -> 7 -> 4 -> 6 -> 5 -> 10 -> 9 -> 8 -> 1	5916

Table 3. Ant Routes in Iteration 3

Ant	Routes (v)	Distance(meters)
Ant 1	1 -> 2 -> 5 -> 3 -> 7 -> 4 -> 6 -> 10 -> 9 -> 8 -> 1	5712
Ant 2	1 -> 2 -> 5 -> 6 -> 4 -> 7 -> 3 -> 9 -> 10 -> 8 -> 1	6027
Ant 3	1 -> 2 -> 5 -> 6 -> 4 -> 7 -> 9 -> 10 -> 8 -> 3 -> 1	6070
Ant 4	1 -> 2 -> 5 -> 3 -> 7 -> 4 -> 6 -> 9 -> 10 -> 8 -> 1	5848
Ant 5	1 -> 2 -> 3 -> 5 -> 6 -> 4 -> 7 -> 8 -> 9 -> 10 -> 1	5709
Ant 6	1 -> 2 -> 5 -> 3 -> 7 -> 4 -> 6 -> 10 -> 9 -> 8 -> 1	5712
Ant 7	1 -> 2 -> 3 -> 5 -> 6 -> 4 -> 7 -> 8 -> 9 -> 10 -> 1	5709
Ant 8	1 -> 2 -> 5 -> 3 -> 7 -> 4 -> 6 -> 10 -> 9 -> 8 -> 1	5712
Ant 9	1 -> 2 -> 3 -> 5 -> 6 -> 4 -> 7 -> 8 -> 9 -> 10 -> 1	5709
Ant 10	1 -> 2 -> 5 -> 3 -> 7 -> 4 -> 6 -> 10 -> 9 -> 8 -> 1	5712

Table 4. Ant Routes in Iteration 4

Ant	Routes (v)	Distance(meters)
Ant 1	1 -> 2 -> 3 -> 5 -> 6 -> 4 -> 7 -> 9 -> 8 -> 10 -> 1	5829
Ant 2	1 -> 2 -> 5 -> 3 -> 7 -> 4 -> 6 -> 10 -> 9 -> 8 -> 1	5712
Ant 3	1 -> 2 -> 5 -> 3 -> 7 -> 4 -> 6 -> 10 -> 9 -> 8 -> 1	5712
Ant 4	1 -> 2 -> 5 -> 3 -> 7 -> 4 -> 6 -> 10 -> 9 -> 8 -> 1	5712
Ant 5	1 -> 2 -> 3 -> 7 -> 4 -> 6 -> 10 -> 9 -> 8 -> 5 -> 1	6091
Ant 6	1 -> 2 -> 5 -> 3 -> 7 -> 4 -> 6 -> 10 -> 9 -> 8 -> 1	5712
Ant 7	1 -> 2 -> 3 -> 5 -> 6 -> 4 -> 7 -> 9 -> 8 -> 10 -> 1	5829
Ant 8	1 -> 2 -> 5 -> 3 -> 7 -> 4 -> 6 -> 10 -> 9 -> 8 -> 1	5712
Ant 9	1 -> 2 -> 5 -> 3 -> 7 -> 4 -> 6 -> 10 -> 9 -> 8 -> 1	5712
Ant 10	1 -> 2 -> 5 -> 3 -> 7 -> 4 -> 6 -> 10 -> 9 -> 8 -> 1	5712

Table 5. Ant Routes in Iteration 5

Ant	Routes (v)	Distance(meters)
Ant 1	1 -> 2 -> 5 -> 4 -> 6 -> 7 -> 3 -> 9 -> 8 -> 10 -> 1	6139
Ant 2	1 -> 2 -> 5 -> 3 -> 7 -> 4 -> 6 -> 10 -> 9 -> 8 -> 1	5712
Ant 3	1 -> 2 -> 5 -> 3 -> 7 -> 6 -> 4 -> 9 -> 8 -> 10 -> 1	5843
Ant 4	1 -> 2 -> 3 -> 5 -> 6 -> 4 -> 7 -> 8 -> 9 -> 10 -> 1	5709
Ant 5	1 -> 2 -> 5 -> 3 -> 7 -> 4 -> 6 -> 10 -> 9 -> 8 -> 1	5712
Ant 6	1 -> 2 -> 3 -> 7 -> 4 -> 6 -> 5 -> 10 -> 9 -> 8 -> 1	5916
Ant 7	1 -> 2 -> 5 -> 3 -> 7 -> 4 -> 6 -> 10 -> 9 -> 8 -> 1	5712
Ant 8	1 -> 2 -> 3 -> 7 -> 4 -> 6 -> 5 -> 8 -> 9 -> 10 -> 1	6044
Ant 9	1 -> 2 -> 5 -> 3 -> 7 -> 4 -> 6 -> 10 -> 9 -> 8 -> 1	5712
Ant 10	1 -> 2 -> 3 -> 5 -> 6 -> 4 -> 7 -> 8 -> 9 -> 10 -> 1	5709

Based on the final simulation results, the ACO algorithm successfully generated an optimal waste collection route in a single trip. All waste collection points were efficiently served by a single vehicle with sufficient capacity.

As part of the algorithm's performance evaluation, tests were conducted on several key parameter combinations in Ant Colony Optimization (ACO), namely α , β , and ρ . The purpose of this testing was to determine the parameter configuration that produces the most efficient waste collection route, in terms of total travel distance and the number of trips formed. Several parameter combinations were tested systematically, while keeping the number of ants at 10 and

the maximum number of iterations at 5. The value of α was varied in the range of 0.5 to 2.0 to assess the influence of pheromone intensity on ant behavior. The value of β was varied from 2.0 to 10 to examine the role of visibility (inverse of distance) in route selection. Meanwhile, ρ or pheromone evaporation rate was tested at values of 0.1, 0.3, and 0.5 to evaluate the stability of the ants' pheromone memory. The following are the results of the parameter testing.

The best route found by the algorithm is shown as follows: Trip 1: $v_1 \rightarrow v_2 \rightarrow v_3 \rightarrow v_5 \rightarrow v_6 \rightarrow v_4 \rightarrow v_7 \rightarrow v_8 \rightarrow v_9 \rightarrow v_{10}$ and then returning to v_1 with a total travel distance of 5709 meter. This route demonstrates that the algorithm is capable of optimizing the sequence of visits between points to minimize the total distance, while also considering the connectivity between nodes in a sparse graph. With only a single trip required, this route exhibits maximum operational efficiency and highlights the successful implementation of ACO in the context of locally data-driven waste transportation.

DISCUSSION

This study demonstrates that the Ant Colony Optimization (ACO) algorithm can be effectively applied to design efficient waste collection routes within a limited simulation setting. However, it is important to note that the simulation was conducted using only 10 waste collection points, representing a simplified version of the actual conditions in the Sekar Tunjung residential area.

In reality, Sekar Tunjung is a considerably larger residential area, with the potential for a significantly greater number of collection points as the number of households continues to grow. Consequently, the waste collection system implemented in such a setting would require a more complex approach, including advanced graph modeling, area segmentation strategies, and adaptation to varying truck capacities and daily waste volumes.

Moreover, the number of ants and iterations used in this simulation was relatively limited (10 ants and 5–50 iterations). In future development, these parameters can be dynamically adjusted based on the size and complexity of the graph. Increasing the number of ants and iterations could enhance solution exploration and improve convergence to optimal routes, although at the cost of higher computational load.

Therefore, this study remains exploratory in nature and presents opportunities for further testing on a larger scale. Future work may include integration with real-time spatial data, waste volume prediction using IoT technologies, and the development of adaptive decision-making systems for sustainable waste management.

CONCLUSION

This study demonstrates that the Ant Colony Optimization (ACO) algorithm is capable of producing efficient solutions for waste collection route planning at a local scale. Through simulations involving 10 collection points in the Sekar Tunjung residential area, ACO successfully generated an optimal route that minimizes total travel distance while considering truck capacity and the waste volume at each point. The results affirm that ACO is an adaptive and effective approach for data-driven waste logistics systems. The benefits of this research include enhanced waste collection efficiency, reduced fuel consumption, and improved urban logistics planning. The proposed method is applicable to other routing-based transportation systems that operate under capacity and spatial constraints.

However, this study is limited by the small number of simulated points and does not yet account for real-time conditions such as traffic congestion, daily waste volume fluctuations, or service time constraints. Therefore, future work is recommended to incorporate real-time data, expand the number of collection points, increase the number of ants and iterations in simulation, and explore hybrid algorithms to achieve more optimal solutions in complex scenarios.

ACKNOWLEDGMENT

This research was supported by Universitas Bali Dwipa through internal research funding. The authors would like to thank the local authorities of Sekar Tunjung Residential Area and TPS3R Kesiman for providing access to spatial and operational data used in this study.

REFERENCES

- Algethami, H. (2023). Performance Evaluation of Ant Colony Optimization in Municipal Waste Collection Systems. *International Journal of Operations Research*, 10(2), 145–159. <https://doi.org/10.1108/ijor.v10i2.2023>
- Arimbawa K., I.B.K.P. (2024). Algoritma Dijkstra : Rute Pengungsian Terpendek Daerah Rawan Bencana di Desa Cangu. *Jurnal Matematika*, 14(1), 52–60. doi: <https://doi.org/10.24843/JMAT.2024.v14.i01.p171>
- Arimbawa K., I.B.K.P., Sukartiasih, W., & Sedayu, A. (2023). Optimization of the Shortest Tsunami Evacuation Route Using Dijkstra's Algorithm in Benoa Village. *Brilliance: Research of Artificial Intelligence*, 3(2), 217–224. <https://doi.org/10.47709/brilliance.v3i2.3089>
- Badan Pusat Statistik Kota Denpasar. (2024). Kota Denpasar Dalam Angka 2024 (Denpasar Municipality in Figures 2024). Denpasar: BPS Kota Denpasar. ISBN: 2338-9303. Diakses dari: <https://denpasarkota.bps.go.id>
- Dorigo, M., & Di Caro, G. (1999). The Ant Colony Optimization Meta-Heuristic. In D. Corne, M. Dorigo, & F. Glover (Eds.), *New Ideas in Optimization* (pp. 11–32). McGraw-Hill.



- Dorigo, M., Maniezzo, V., & Colorni, A. (1996). Ant System: Optimization by a colony of cooperating agents. *IEEE Transactions on Systems, Man, and Cybernetics - Part B: Cybernetics*, 26(1), 29–41. <https://doi.org/10.1109/3477.484436>
- Gunawan, A., Putra, R., & Santoso, D. (2024). Implementasi Algoritma Ant Colony Optimization untuk Sistem Pengangkutan Sampah Kota Tegal. *Jurnal Sistem Informasi dan Rekayasa*, 12(1), 45–53.
- Hu, Y., Ju, Q., Peng, T., Zhang, S., & Wang, X. (2024). Municipal solid waste collection and transportation routing optimization based on IAC-SFLA. *Journal of Environmental Engineering and Landscape Management*, 32(1), 31–44. <https://doi.org/10.3846/jeelm.2024.20774>
- Hu, Z., Li, Q., & Wang, H. (2024). Hybrid IAC-SFLA Algorithm for Optimizing Domestic Waste Collection Routes. *Waste Management & Research*, 42(1), 12–25. <https://doi.org/10.xxxx/wmr.2024.42.1>
- Ky Phuc, P. N., & Phuong Thao, N. L. (2021). Ant Colony Optimization for Multiple Pickup and Multiple Delivery Vehicle Routing Problem with Time Window and Heterogeneous Fleets. *Logistics*, 5(2), 28. <https://doi.org/10.3390/logistics5020028>
- LCDI Indonesia. (2024). Peta Jalan Pengelolaan Sampah Menuju Nol Sampah 2040. Jakarta: Low Carbon Development Initiative Indonesia.
- Manoharan, M., Wu, Q., & Li, Z. (2025). Smart Solid Waste Management Using Swin Transformer and Ant Colony Optimization. *IEEE Internet of Things Journal*, 12(3), 3456–3468. <https://doi.org/10.xxxx/iotj.2025.12.3>
- Okrah, M. B., Owusu, G. A., & Kwakye, K. (2023). Ant Colony Optimization for Waste Collection in Urban Ghana: A Sustainability Perspective. *African Journal of Environmental Science*, 19(2), 73–88. <https://doi.org/10.xxxx/ajes.2023.19.2>
- Rattanawai, S., Namsang, S., & Tungsong, C. (2024). Optimization of Medical and Non-Medical Waste Collection Routes using Ant Colony Algorithm. *Journal of Healthcare Engineering*, 2024, Article ID 345923. <https://doi.org/10.1155/2024/345923>
- Shen, L., Zhang, P., & Chen, Y. (2023). Comparison of Ant Colony and Particle Swarm Optimization for Multi-trip Urban Waste Routing. *Journal of Cleaner Logistics*, 11(4), 89–101. <https://doi.org/10.xxxx/jcl.2023.11.4>
- Wouda, M., van Acker, T., & Braet, J. (2024). Integrated Decision Support for Waste Collection and Routing Using Data-Driven ACO. *Sustainable Cities and Society*, 83, 103907. <https://doi.org/10.1016/j.scs.2024.103907>