Optimization Of Vehicle Routing Problem With Time Windows (VRPTW) Using Hybrid Ant Colony Optimization (ACO) On Food Product Distribution Routes

Bonanda Siregar[1], Yunita[2]
[1,2,]Informatics Engineering, Faculty Of Computer Science, Sriwijaya University, Indonesia
E-mail: [1]**Bonandasiregar40@gmail.com**, [**2]yunita.v1t4@gmail.com

Abstract

The problem of vehicle scheduling and route optimization in food product distribution is a complex logistical challenge, especially when considering customer time window constraints and vehicle capacity. This study aims to implement a combination of Ant Colony Optimization (ACO) and Nearest Neighbor (NN) to solve the Vehicle Routing Problem with Time Windows (VRPTW) in the distribution system of HoneyBee Bakery & Cake. The Nearest Neighbor (NN) method is used to generate the initial solution, while Ant Colony Optimization (ACO) is applied to further optimize the route. The experimental results show that the Hybrid Ant Colony Optimization approach produces an initial solution with a travel distance of 72.55 km, which is then optimized to a final distance of 58.59 km with four returns to the depot. Therefore, the Hybrid ACO is proven to be more effective in optimizing distribution routes that consider both time constraints and vehicle capacity, achieving an efficiency improvement of 19.22% in reducing the total travel distance from the initial solution.

Keywords*—Vehicle Routing Problem with Time Windows, Ant Colony Optimization, Nearest Neighbor*

1. INTRODUCTION

In the logistics sector, efficient distribution is crucial to ensure timely product availability at various locations. In the food industry, the main challenge lies in optimizing delivery routes since products are highly sensitive to time and quality. Inefficient distribution may cause product spoilage and customer dissatisfaction. HoneyBee Bakery & Cake is a micro-enterprise specializing in bakery products such as donuts. It distributes its products daily across Palembang. The company experiences issues with delayed deliveries due to non-optimal route planning, highlighting the need for an intelligent routing solution.

The Vehicle Routing Problem with Time Windows (VRPTW) addresses the optimization of vehicle routes under constraints such as delivery time windows and vehicle capacities. A variant of the Vehicle Routing Problem (VRP), VRPTW is a combinatorial problem that seeks to minimize total travel cost while fulfilling constraints related to time and capacity. This research utilizes a hybrid approach by combining Nearest Neighbor (NN) and Ant Colony Optimization (ACO). NN serves as the initial solution constructor by quickly identifying nearby delivery points, while ACO optimizes these routes by simulating ant behavior using pheromone trails. The objective is to enhance delivery efficiency for HoneyBee Bakery.

According to [1], distribution is the process of delivering goods or services from producers to consumers at the time they are needed. This definition highlights the importance of timely delivery in ensuring customer satisfaction and maintaining product quality. In the food sector, delays in delivery can lead to freshness degradation or spoilage. Thus, a well-optimized routing strategy is essential to meet customer expectations and reduce operational inefficiencies.

2. LITERATURE STUDY

2.1. Distribution

Distribution refers to the process of delivering goods or services from producers to consumers. According to [2], it involves managing goods, determining delivery routes, and controlling delivery timing. In the context of food logistics, efficient distribution ensures products arrive fresh, in good condition, and on time, minimizing waste and maximizing customer satisfaction.

2.2 routing problem

Vehicle Routing Problem (VRP), introduced by Dantzig and Ramser in 1959, is a generalization of the Traveling Salesman Problem (TSP). It is a combinatorial problem where a fleet of vehicles must serve a set of geographically scattered customers while minimizing total travel cost or time. According to [3], Vehicle Routing Problem (VRP) is a combinatorial problem aimed at determining distribution routes that can be applied to commodity distribution mechanisms, with the goal of finding efficient delivery routes. This perspective emphasizes the practical use of VRP in optimizing real-world logistics operations, especially in the distribution of goods or services.

Several variants of VRP have been developed to handle different real-world constraints, such as Capacitated VRP (CVRP), VRP with Pickup and Delivery (VRPPD), VRP with Time Windows (VRPTW), and Multi Depot VRP (MDVRP), each designed to solve specific distribution challenges in logistics systems.

2.3 Vehicle routing problem with time windows

Vehicle Routing Problem with Time Windows (VRPTW) is a well-known optimization problem that focuses on finding the optimal routes for a fleet of vehicles to serve customers within specified time windows, with the objective of minimizing the total cost or distance traveled [4].

According to [5], the main objective is to minimize the number of vehicles and total travel distance while meeting service time constraints. The mathematical model includes constraints such as vehicle capacity, service time alignment, and arrival within time windows.

Generally, there are two objectives in VRPTW, which are defined as Eqs. (1) and (2), respectively. The primary goal is to minimize the number of vehicles (NV) and the secondary goal is to minimize the total distance (TD) with the same number of routes. VRPTW can be mathematically formulated as follows.

$$\min NV = K \tag{1}$$

$$\min TD = \sum_{i=0}^{M}\sum_{j=0}^{M}\sum_{k=1}^{K} c_{i,j} * x_{i,j}^{k} \tag{2}$$

s.t.

$$\sum_{i=0}^{M} x_{i,j}^{k} = y_{j}^{k} \quad \forall k = 1, \dots, K, \quad \forall j = 1, \dots, M \tag{3}$$

$$\sum_{j=0}^{M} x_{i,j}^{k} = y_{i}^{k} \quad \forall k = 1, \dots, K, \quad \forall i = 1, \dots, M \tag{4}$$

$$\sum_{k=1}^{K} y_{i}^{k} = 1 \quad \forall i = 1, \dots, M \tag{5}$$

$$\sum_{i=0}^{M} y_i^k * q_i \leq Q \quad \forall k = 1, \dots, K \tag{6}$$

$$\sum_{k=1}^{K} y_0^k = K \tag{7}$$

$$t_i + w_j + s_i + c_{i,j} = t_j \quad \forall i, j = 1, \dots, M, i \neq j \tag{8}$$

$$b_j \leq t_j \leq l_j \quad \forall j = 1, \dots, M \tag{9}$$

$$w_j = \max\{b_i - (t_i + s_i + c_{i,j}), 0\} \quad \forall i = 1, \dots, M \tag{10}$$

Where

$$x_{i,j}^k = \begin{cases} 1, if\ vehicle\ k \text{ travels directly from } i \text{ to i} \\ 0, \text{otherwise} \end{cases}$$

$$x_{i,j}^k = \begin{cases} 1, if\ vehicle\ k \text{ travels directly from } i \text{ to i} \\ 0, \text{otherwise} \end{cases}$$

Constraints Eqs. (3)–(5) mean that each customer will be served by a vehicle and that each customer can be served by only one vehicle. Constraint Eq. (6) means that each vehicle cannot carry more than capacity Q. Constraint Eq. (7) denotes that all routes start from the depot. Eqs. (8)–(10) define the time window constraint, where ti is the vehicle arrival time at node i; wj is the waiting time after the vehicle arrives at customer j; Si is the service time; and ci,j is the time cost between nodes i and j.

2.4 Metaheuristic

Metaheuristic algorithms are search methods inspired by social behavior or natural principles. Unlike traditional algorithms, they combine random processes with local search. The two main parts are exploration, which means searching widely for different solutions, and exploitation, which means focusing on improving good solutions in a local area [6]. According to [7] metaheuristics are computational approaches used to find near-optimal solutions to optimization problems by iteratively improving candidate solutions while considering solution quality constraints. This highlights the strength of metaheuristics in solving complex problems where exact methods are computationally impractical, especially in large-scale combinatorial optimization cases such as vehicle routing.

2.5 Swarm Intelligence Algorithm

Swarm Intelligence Algorithm is a method inspired by how groups of animals like birds, fish, or insects work together in nature. The process starts with an initial setup, then each agent is evaluated using a fitness function. Based on this, agents are updated and guided to find better solutions until a stopping condition is met. In the end, the best solution found is taken as the final result [8]. These algorithms are widely used in optimization because they are good at finding the best solutions efficiently. Some popular swarm intelligence algorithms include:

1. **Ant Colony Optimization (ACO):** Created by Marco Dorigo in 1992, ACO is based on how ants find the shortest path to food by following pheromone trails.

2. **Particle Swarm Optimization (PSO):** Inspired by how birds fly in groups to find food. Each "particle" is a possible solution that moves and learns from its own and the group's best experiences.

3. **Dragonfly Algorithm (DA):** Based on how dragonflies balance between exploring new areas and using known good areas to find food or avoid danger.

4. **Artificial Fish Swarm (AFS):** Imitates how fish search for food, gather in groups, and follow others to find better areas, aiming to find the best solution.

5. **Artificial Bee Colony (ABC):** Introduced by Karaboga in 2005, this algorithm copies how honey bees search for food, with different roles like worker bees, onlooker bees, and scout bees, to find the best solution.

2.6 Nearest neighbor

The Nearest Neighbor method is an algorithm used in the Vehicle Routing Problem (VRP) to determine delivery routes by selecting the nearest customer as the next destination from the current service point. This method is a simple and fast approach to solving VRP; however, it does not guarantee an optimal solution as decisions are made solely based on the nearest customer at each step [9].

2.7 Ant Colony Optimization

In the early 1990s, M. Dorigo and his team introduced Ant Colony Optimization (ACO), a method inspired by how ants find food. When ants find food, they leave a pheromone trail on the way back to the nest. Other ants follow this trail, and if the path is short and used often, the pheromone becomes stronger. This helps more ants choose the best path. If a path is not used much, the pheromone fades, and ants stop using it.

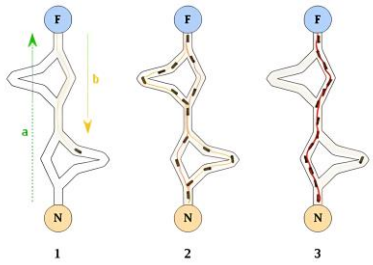


Figure 1. Illustration ants finding food

The more ants use a path, the more likely other ants will follow it. According to [4], here are the steps to solve problems using Ant Colony Optimization and the Vehicle Routing Problem with Time Windows:

1. Randomly choose nodes from depot.
2. Find the highest probability nodes from nodes i to nodes j. Check the distance, time and capacity constraints.

$$P_k(i,j) = \begin{cases} \dfrac{[\tau(i,j)]^{\alpha}[\eta(i,j)]^{\beta}}{\sum_{\text{unvisited customer}}[\tau(i,j)]^{\alpha}[\eta(i,j)]^{\beta}} & jika\ i \neq j \\ atau\ 0 \quad jika\ i = \text{j} \end{cases}$$

3. Update pheromone value

$$P_k(i,j) = \begin{cases} \dfrac{[\tau(i,j)]^{\alpha}[\eta(i,j)]^{\beta}}{\sum_{\text{unvisited customer}}[\tau(i,j)]^{\alpha}[\eta(i,j)]^{\beta}} & jika\ i \neq j \\ atau\ 0 \quad jika\ i = \text{j} \end{cases}$$

4. Find the next highest probability nodes for unvisited nodes and construct a vehicle route.
5. Calculate distance between two customer
6. Construct the final solution.

Ant Colony Optimization (ACO) for solving the Vehicle Routing Problem with Time Windows (VRPTW) is done using the following simple steps:.

1. Initialization: Set VRPTW parameters like customers, time windows, vehicle capacity, and depot. Place one ant for each vehicle and choose a random starting node from the depot.

2. Ant Movement: Each ant moves from the depot and picks the next customer based on probability, making sure the route still follows capacity and time window limits. This continues until all customers are visited or no more can be added.

3. Pheromone Update: After finishing their routes, ants leave pheromones on the paths they used. More pheromone means a higher chance other ants will follow the same path next time.

4. Iteration: Repeat the ant movements and pheromone updates several times or until a stopping condition is reached, to keep improving the route solution.

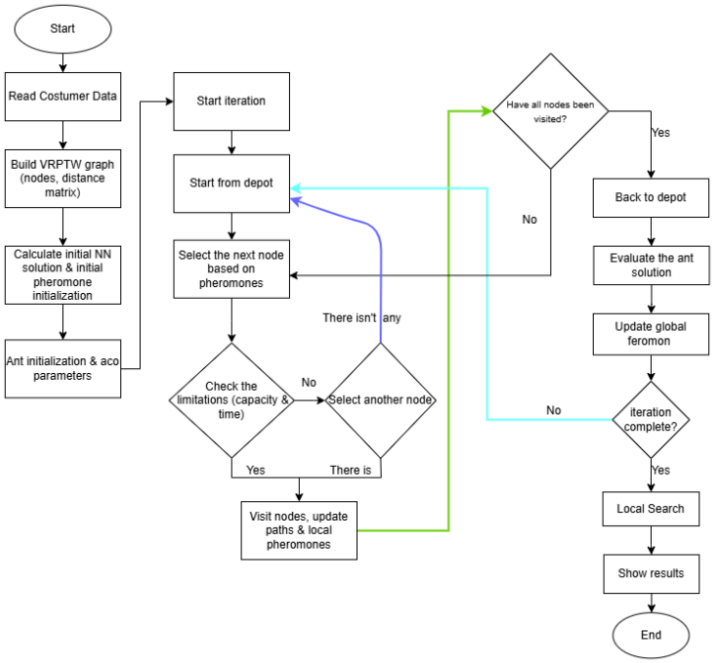


Figure 2. Flowchart Ant colony optimization + Nearest neighbor

2.8 Efficiency

According to [10] efficiency can be defined as optimizing and maximizing an activity or operation appropriately and accurately in order to save or minimize costs, time, and effort. To measure efficiency, the initial distance is compared to the optimized distance, and the result is expressed as a percentage of savings. The following is the efficiency formula:

$$\text{Efficiensy} = \frac{Initial\ Distance - \ Optimized\ Distance}{Initial\ Distance} \times 100$$

3. RESEARCH METHODS

3.1. Research Workflow

The research workflow consists of several stages: literature study, data collection, algorithm design, implementation, and testing. First, a literature review was carried out to understand different types of Vehicle Routing Problems (VRP) and hybrid optimization methods. After that, the collected data were processed to define constraints and customer distributions.

The algorithm design combines Nearest Neighbor (NN) and Ant Colony Optimization (ACO). Nearest Neighbor is used to create an initial solution. This initial route is then optimized using the ACO algorithm, which simulates ant behavior to search for better solutions by exploring various routes, updating pheromone trails, and minimizing total travel distance while meeting vehicle capacity and time window constraints.

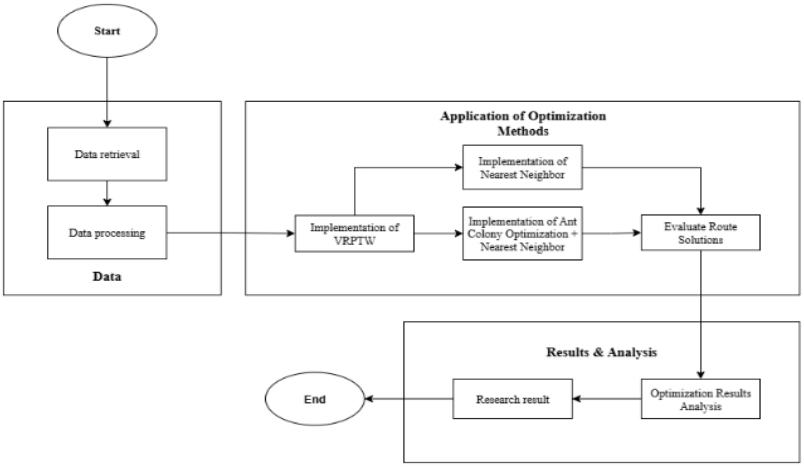


Figure 3. Research Workflow

3.2. Data Collection

This study uses primary data collected from HoneyBee Bakery & Cake. The data includes customer order information such as customer names, locations, distances between locations, number of products ordered, delivery schedule, and vehicle capacity. After collecting the data, it is processed and reformatted to match the needs of the analysis and the Ant Colony Optimization algorithm.

Table 1. Table Customer Location

Id	X	Y	Demand	Ready time	Due time
0	-2.967797	104.726648	0	00:00	00:00
1	-2.980148	104.755182	112	04:00	06:00
2	-2.953332	104.735306	70	04:00	05:00
3	-2.915004	104.680069	40	03:00	05:00
4	-2.997204	104.707322	90	05:00	06:00
5	-2.957643	104.726935	60	06:00	07:00
6	-2.993812	104.726361	30	06:00	07:00
7	-2.955790	104.748410	30	06:00	07:00
8	-2.967871	104.746621	25	06:00	08:00
9	-2.962252	104.749706	25	06:00	08:00

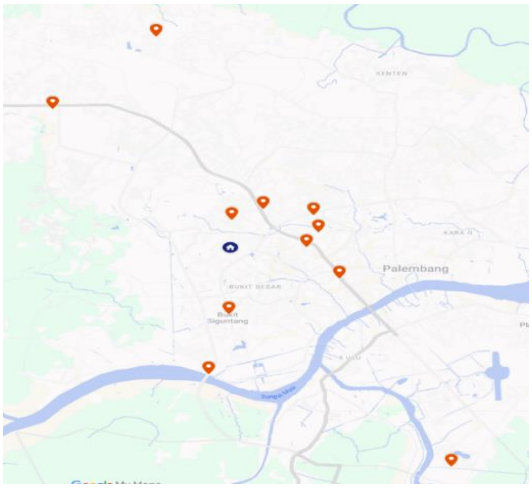


Figure 4. Customer Location

4. RESULTS AND DISCUSSION

The developed application includes several key features such as a file upload option, display of the initial solution, display of the final optimized solution, and settings to adjust ACO parameters. The interface showcasing these features is presented in Figure.5.

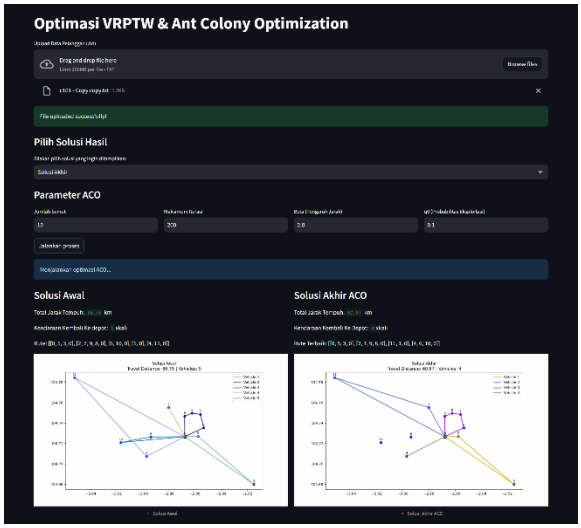


Figure 5. System Interface

4.1. Trial configuration

The experiment uses customer data collected from HoneyBee Bakery & Cake. The dataset includes customer location coordinates, demand quantity, and service time, provided in .txt format and used as input for the route optimization process.

In this research, the Nearest Neighbor and Ant Colony Optimization (ACO) methods are used to solve the Vehicle Routing Problem with Time Windows (VRPTW). The experiment focuses on testing different ACO parameter settings to find the best routes with the shortest total distance and the minimum number of vehicles returning to the depot. The experiment includes different test scenarios with variations in key ACO parameters: number of ants (100, 50,20, 10 ,5), number of iterations, β value, and Q value.

4.2. Test Result

Table 2. Table Result Test 1

Test	1	2
Number Ant	100	100
Max Iteration	200	200
Beta	1.0	1.0
Q	2	1
Total Distance	75.77 km	75.77 km
Vehicle Returning to Depot	6 times	6 times

Table 3. Table Result Test 2

Test	1	2	3	4	5
Number Ant	100	50	20	10	5
Max Iteration	200	200	200	200	200
Beta	1.0	1.0	1.0	1.0	1.0
Q	0.5	0.5	0.5	0.5	0.5
Total Distance	59,93km	59,93km	60.27km	61.02km	59.68km
Vehicle Returning to Depot	4 times	4 times	4 times	4 times	4 times

Table 4. Table Result Test 3

Test	1	2	3	4	5
Number Ant	100	50	20	10	5
Max Iteration	200	200	200	200	200
Beta	1.0	1.0	1.0	1.0	1.0
Q	0.3	0.3	0.3	0.3	0.3
Total Distance	59.93km	58.59km	59.93km	61.02km	60.76km
Vehicle Returning to Depot	4 times	4 times	4 times	4 times	4 times

Table **5.** Table Result Test 4

Test	1	2	3	4	5
Number Ant	100	50	20	10	5
Max Iteration	200	200	200	200	200
Beta	1.0	1.0	1.0	1.0	1.0
Q	0.1	0.1	0.1	0.1	0.1
Total Distance	58.59km	58.59km	58.59km	58.59km	59.63km
Vehicle Returning to Depot	4 times	4 times	4 times	4 times	4 times

The test results in Table II, using a Q value of 2 and 1, with 100 ants, 200 iterations, and a beta value of 1, showed a total travel distance of 75.77 km. The vehicles returned to the depot 6 times, and the routes remained the same. A high Q value like this causes the ants to quickly focus on paths with strong pheromones, without enough exploration of other possible routes. As a result, the final solution is less optimal.

In Table III, with a Q value of 0.5, the number of ants was varied at 100, 50, 20, 10, and 5. The results showed that the total travel distance remained relatively high. The first and second tests (100 and 50 ants) produced a distance of 59.93 km. When the number of ants was reduced to 20 and 10, the distance increased to 60.27 km and 61.02 km. However, in the fifth test with only 5 ants, the result slightly improved to 59.68 km. This happened because fewer ants meant less exploration, but the result may not be optimal.

In Table IV, with a Q value of 0.3, the algorithm's performance was more varied. The second test with 50 ants gave the best result of 58.59 km. Other tests showed distances ranging from 59.93 km to 61.02 km. This shows that the combination of Q value and number of ants affects performance. If there are enough ants, the algorithm can explore the solution space well. But when there are too few ants, the exploration becomes limited, and the solution quality may drop due to a lack of path variation.

Lastly, in the test with the lowest Q value, 0.1, the results were very stable and optimal. Tests with 100, 50, 20, and 10 ants all produced a distance of 58.59 km. This shows that a low Q value helps ants explore more routes and makes it easier to find the best solution.

But with only 5 ants, performance dropped again with a distance of 59.63 km, showing that too few ants still limit exploration, even if the Q value is ideal.

In conclusion, the best performance was achieved with a low Q value (e.g., 0.1) and a large number of ants (e.g., 100 or 50). A low Q value supports wide exploration, and enough ants ensure diverse solutions. On the other hand, a high Q value (1 or 2) often gives poor results because the ants focus too much on early solutions and miss better routes.

To evaluate the performance of the Hybrid Ant Colony Optimization (ACO) algorithm, a comparison was made with the standard Ant Colony Optimization (ACO) without Nearest Neighbor (NN). The focus of the comparison is on travel distance efficiency and distribution effectiveness, measured by the number of vehicles that must return to the depot. Table VI presents the performance comparison between the ACO and ACO+NN algorithms using the best configuration: 100 ants, 200 maximum iterations, Beta = 1, and Q = 0.1.

Table 6. Table Comparasion

Method	*Ant Colony Optimization*	*Ant Colony Optimization + Nearest Neighbor*
Initial distance	64.09 km	72.55 km
Optimize distance	58.59 km	58.59 km
Iteration Required	46 iteration	10 iteration
Vehicle Returning to Depot	4 times	4 times

$$\text{Efficiency} = \left(\frac{72{,}55 - 58{,}59}{72{,}55}\right) \times 100 = 19.22\%$$

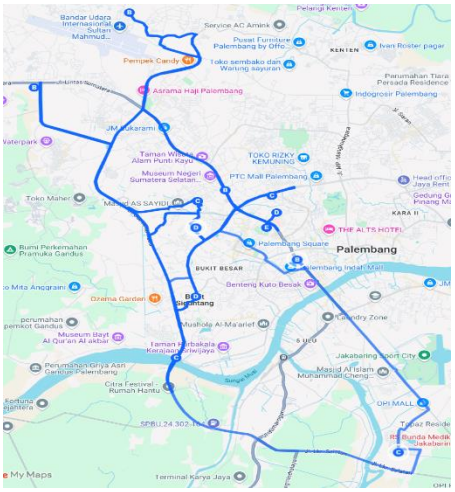


Figure 6. Customer Routes

The test results show that even though the pure ACO method started with a shorter initial route than the hybrid ACO, it needed more time to reach the best result, taking 46 iterations. On the other hand, ACO+NN found the best solution much faster, in just 10 iterations. This shows that the structured starting solution from Nearest Neighbor (NN) helps speed up the search process. Both methods reached the same final distance of 58.59 km, but the hybrid method was more efficient in terms of computation time. with an efficiency increase of 19.22% from the initial solution.

5. CONCLUSION

1. The use of the Hybrid Ant Colony Optimization (ACO) algorithm has proven effective in solving the VRPTW problem for food product distribution. Hybrid ACO successfully handles time window and capacity constraints. It achieved the shortest travel distance of 58.59 km with the best parameter settings: 100 ants, 200 iterations, $\beta = 1.0$, and $Q = 0.1$, with a travel distance improvement of 19.22% compared to the initial solution.

2. Hybrid ACO is also efficient in minimizing the number of vehicle returns to the depot. It only required 4 returns to complete all deliveries and did so with fast optimization. This shows that Hybrid ACO optimizes vehicle capacity effectively and offers extra benefits in efficiency process.

3. Parameter testing of the Hybrid ACO algorithm shows that the number of ants and the exploitation probability (Q) value greatly affect the solution quality. More ants and smaller Q values produce better routes. Applying the Hybrid ACO to HoneyBee Bakery & Cake's distribution system had a positive impact. The results showed fewer depot returns and shorter travel distances, helping to reduce travel time and improve customer service

6. SUGGESTED

Suggestions for Future Research:

1. Conduct research using other metaheuristic algorithms such as Particle Swarm Optimization (PSO) and Genetic Algorithm (GA) to evaluate the advantages of each algorithm.
2. Integrate the route optimization system into a mobile application.
3. Develop the system to display visualization results in map form, similar to Google Maps.

7. REFERENCES

[1] M. T. Muslim, "PENGARUH BIAYA PROMOSI DAN BIAYA DISTRIBUSI TERHADAP LABA BERSIH PADA PT. UNILEVER INDONESIA Tbk. PERIODE 2006-2013," Oct. 2020.

[2] M. Rafi, F. Okana, G. Alethea, and A. Devi, "OPTIMALISASI PEMILIHAN RUTE DAN MODA PADA PROSES DISTRIBUSI PRODUK YANG KAITANNYA DENGAN BIAYA LOGISTIK," 2024.

[3] E. Setyati and I. Juniwati, "Ant Colony Optimization untuk Menyelesaikan Perutean Distribusi Snack dengan Vehicle Routing Problem," 2022. [Online]. Available: https://doi.org/10/25047/jtit.v9i2.296

[4] Y. Y. Chai and F. Johar, "Optimizing Blood Transport Costs: Ant Colony Optimization Method for Vehicle Routing Problem with Time Windows," 2024. [Online]. Available: https://science.utm.my/procscimath/

[5] Q. Wu *et al.*, "A neighborhood comprehensive learning particle swarm optimization for the vehicle routing problem with time windows," *Swarm Evol Comput*, vol. 84, Feb. 2024, doi: 10.1016/j.swevo.2023.101425.

[6] R. Hu, L. Bao, H. Ding, D. Zhou, and Y. Kong, "Analysis of the influence of population distribution characteristics on swarm intelligence optimization algorithms," *Inf Sci (N Y)*, vol. 645, Oct. 2023, doi: 10.1016/j.ins.2023.119340.

[7] E. Febianti, Y. Muharni, D. Falti, L. Herlina, and K. Kulsum, "Usulan Penjadwalan Mesin Paralel Menggunakan Metode Ant Colony Optimization Algorithm dan Longest Processing Time," *Journal of Integrated System*, vol. 6, no. 1, pp. 42–52, Jul. 2023, doi: 10.28932/jis.v6i1.5610.

[8] J. Tang, G. Liu, and Q. Pan, “A Review on Representative Swarm Intelligence Algorithms for Solving Optimization Problems: Applications and Trends,” Oct. 01, 2021, *Institute of Electrical and Electronics Engineers Inc.* doi: 10.1109/JAS.2021.1004129.

[9] P. Alamsyah and J. Arifin, “Analisis Pendistribusian Produk Kepada Konsumen Menggunakan Metode Nearest Neighbor di PT. Bukit Muria Jaya,” 2023.

[10] M. Ridwan and M. Rizal Gaffar, “EFISIENSI PERSEDIAAN DAN DISTRIBUSI MELALUI INTEGRASI SUPPLY CHAIN MANAGEMENT,” 2022.