Edu Komputika 13 (1) (2026)

Edu Komputika Journal

https://journal.unnes.ac.id/journals/edukom/

Tourism Route Optimization Based on the Travelling Salesman Problem Using Ant Colony Optimization in the Banyumas and Purbalingga Regions

Teddy Putratama and Sandhy Fernandez✉

Information Systems, Telkom University, Purwokerto, 53147, Indonesia

Article Info

Article History:
Received: 16 May 2026
Revised: 13 August 2026
Accepted: 29 August 2026

Keywords:
Ant Colony Optimization Algorithm, Route Optimization, Tourism Route, Travelling Salesman Problem.

Abstract

Tourism route planning requires an efficient visiting sequence to reduce travel distance and transportation costs. When multiple tourist destinations must be visited in a single trip, the route-planning problem can be formulated as a Travelling Salesman Problem (TSP). This study implements Ant Colony Optimization (ACO) to determine an efficient route among 12 tourist destinations in Banyumas and Purbalingga. The destinations were represented as nodes in a weighted graph, while the travel distance between each pair of destinations was used as the edge weight. The algorithm constructed candidate routes using probabilistic selection and updated pheromone values based on route quality. Three pheromone evaporation rates, namely $\rho=0.3$, $\rho=0.5$, and $\rho=0.7$, were evaluated. The shortest route obtained in the first iteration was W1–W3–W2–W4–W8–W9–W7–W10–W11–W12–W6–W5–W1, with a total distance of 109.0 km. After pheromone updating, the pheromone levels on the shortest-route edges were 0.016174, 0.014174, and 0.012174 for evaporation rates of 0.3, 0.5, and 0.7, respectively. These results show that the evaporation rate directly affects pheromone retention, with a lower evaporation rate maintaining stronger pheromone information on the selected route. The study demonstrates that ACO can generate an efficient tourism route and provides a computational basis for developing a tourism route decision-support system.

✉ Correspondence Address:
Jl. DI Panjaitan No.128, Karangreja, Purwokerto Kidul,
Kec. Purwokerto Sel., Kabupaten Banyumas, Jawa Tengah 53147
E-mail: sandhyf@telkomuniversity.ac.id

ISSN 2252-6811
E-ISSN 2599-297X

INTRODUCTION

Tourism travel route planning plays an important role in improving travel time and distance efficiency, especially when tourists are required to visit multiple destinations within a single trip. The Banyumas and Purbalingga regions have numerous tourist destinations that are geographically dispersed, so an inefficient determination of the visiting sequence can increase distance and travel time (Hjeij & Vilks, 2023; Wang & Han, 2021). This problem can be modeled as the Traveling Salesman Problem, which is a combinatorial optimization problem that aims to determine the shortest route that visits all locations exactly once and returns to the starting point (Ochelska-Mierzejewska et al., 2021; Wang & Han, 2021; Zeng et al., 2021). The Traveling Salesman Problem is classified as a non-deterministic polynomial-time hard problem, making exact approaches computationally inefficient as the number of nodes increases. Therefore, heuristic and metaheuristic approaches are required to obtain near-optimal solutions within a reasonable computational time (Goel et al., 2025; Linganathan & Singamsetty, 2024; Nourmohammadzadeh & Voß, 2026).

The Ant Colony Optimization algorithm is a metaheuristic method inspired by the behavior of ant colonies in finding the shortest paths to food sources (M. Almufti et al., 2023; Stützle & Dorigo, 2004; Wang & Han, 2021; Wangying & Naiming, 2025). This algorithm constructs solutions incrementally through indirect interactions among ant agents by using pheromones as a communication medium. The pheromone intensity and distance heuristic values influence the probability of path selection in each iteration, causing routes with better quality to be increasingly reinforced through the pheromone updating mechanism (Blum, 2024; Stützle & Dorigo, 2004). This study analyzes the computational mechanism of the Ant Colony Optimization algorithm for constructing optimal tourism routes, particularly its stochastic path selection via the Roulette Wheel method and the best-only pheromone updating scheme. The pheromone evaporation process helps prevent premature convergence and maintain a balance between exploration and exploitation of paths (Han et al., 2024; Hjeij & Vilks, 2023).

Several previous studies have demonstrated the effectiveness of the Ant Colony Optimization algorithm in solving route optimization problems based on the Traveling Salesman Problem. Armond et al. (Armond et al., 2022) applied Ant Colony Optimization to optimize Trans Banyumas bus routes and showed that an appropriate parameter configuration was able to produce routes with shorter distances and travel times compared to the original routes. Meanwhile, Anggraeni et al. (Anggraeni et al., 2024) compared the performance of Ant Colony Optimization and Simulated Annealing in solving a delivery routing problem in Surabaya and found that both algorithms were able to improve route efficiency, although Simulated Annealing produced a more optimal solution in that case.

This study contributes to tourism route optimization research by applying the Ant Colony Optimization algorithm to a weighted graph model based on distance data between tourist destinations in Banyumas and Purbalingga with fixed algorithm parameters. The research focuses on analyzing the route construction process and the effect of pheromone evaporation rates on the quality of the resulting solutions. This study is limited to computational and simulation aspects without the development of a user-based system or application; therefore, the results are concentrated on evaluating the performance of the algorithm in solving the Traveling Salesman Problem within the context of tourism routes.

Several approaches can be used to solve the Traveling Salesman Problem. Exact methods, such as brute-force enumeration, dynamic programming, branch-and-bound, and integer linear programming, can identify a globally optimal solution. However, their computational requirements generally increase rapidly as the number of destinations grows. Heuristic methods, including nearest neighbor, insertion heuristics, 2-opt, and 3-opt, can produce solutions more quickly, although they may become trapped in locally optimal routes. Population-based metaheuristics, such as Genetic Algorithm, Particle Swarm Optimization, Simulated Annealing, and Ant Colony Optimization, provide alternative search mechanisms for exploring a larger solution space. Each method has different advantages in terms of solution quality, computational cost, convergence behavior, parameter sensitivity, and implementation complexity.

This study formulates tourism route planning as a TSP because the planned journey requires each selected destination to be visited exactly once before returning to the initial destination while minimizing the total travel distance. ACO was selected because its solution-construction mechanism naturally represents a route as a sequence of connected destinations. Artificial ants explore different visiting sequences using distance information and accumulated pheromone values, while pheromone evaporation reduces the continued dominance of previously selected routes. This mechanism supports iterative exploration and reinforcement of shorter routes without enumerating every possible route. ACO also provides a flexible computational framework that can be extended to larger destination sets and additional tourism constraints. Nevertheless, this study does not assume that ACO always outperforms other TSP methods. The method was selected because its route-construction process is compatible with the structure of the tourism routing problem examined in this study.

Accordingly, this study aims to implement ACO for determining an efficient visiting sequence among 12 tourist destinations in Banyumas and Purbalingga and to examine the effect of different pheromone evaporation rates on the search process. The study focuses on the route produced by the configured ACO model rather than claiming that ACO is universally superior to other optimization methods.

RESEARCH METHODS

This study employs a computational and simulation-based approach to analyze the tourism route optimization problem formulated as the Traveling Salesman Problem. This approach is selected to examine the algorithmic mechanisms involved in constructing optimal routing solutions based on distance data between tourist destinations. The research methodology includes stages of distance data collection, weighted graph modeling, the design and implementation of the Ant Colony Optimization algorithm, as well as simulation and result evaluation. All stages are systematically designed to ensure that the solution search process is conducted consistently and aligned with the research objective, namely, to obtain a tourism travel route with minimum total distance through an adaptive computational mechanism.

A. Data and Graph Modeling

This study employs a computational and simulation-based approach to solve the tourism route optimization problem modeled as the Traveling Salesman Problem (Hjeij & Vilks, 2023; Wang & Han, 2021). The Traveling Salesman Problem is a combinatorial optimization problem that aims to determine the shortest route that visits each location exactly once and returns to the starting location (Ochelska-Mierzejewska et al., 2021; Wang & Han, 2021; Zeng et al., 2021). In the context of graph theory, the solution to the Traveling Salesman Problem is represented as a minimum Hamiltonian cycle on a weighted graph, namely a cycle that traverses all vertices without repetition and has the minimum total weight. Determining a minimum Hamiltonian cycle on a weighted graph belongs to the class of NP-Hard problems, making exact solutions computationally inefficient as the number of vertices increases (Bondy & Murty, 1976; Buinoschi-Tirpescu & Breaban, 2024; Chauhan, 2024; Shi & Zhang, 2021; Tao et al., 2023).

The data used in this study consist of distances between tourist destinations in the Banyumas and Purbalingga regions. The distance data were obtained through Google Maps observations by selecting the shortest travel distance between tourist locations. All distance data were then organized into a weighted distance matrix. Each tourist destination is modeled as a vertex, while the distance between locations is modeled as an edge weight. This study uses twelve vertices labeled W1 to W12. The resulting graph is a complete weighted graph, where each vertex is directly connected to every other vertex and all edge weights are positive. This graph model is used as a formal representation of the Traveling Salesman Problem and serves as the primary input for the Ant Colony Optimization algorithm.

The weighted graph model representing the relationships between tourist destinations is used as the basis for searching for a minimum Hamiltonian cycle using the Ant Colony Optimization algorithm. This graph representation illustrates the position of each vertex and the quantitative distance relationships between vertices, as shown in Figure 1.

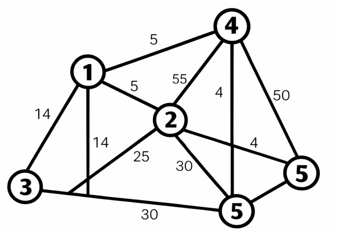

Figure 1. Weighted graph

The tourism route optimization problem was represented as a complete weighted graph, as shown in Figure 2. The graph consists of 12 nodes, denoted by W1 to W12, with each node representing one tourist destination in Banyumas or Purbalingga. The edges represent the available road connections between tourist locations, while their weights indicate the travel distances in kilometers obtained from the distance matrix.

This representation enables the tourism route problem to be formulated as a Traveling Salesman Problem, in which every destination must be visited exactly once before the route returns to its initial destination.

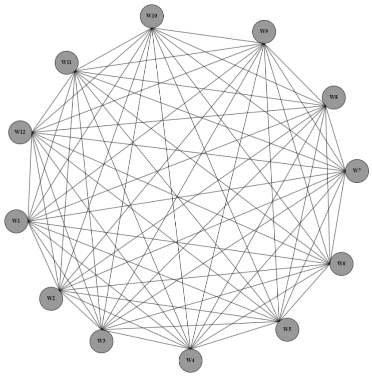

Figure 2. Weighted graph of tourist locations W1-W12 as a traveling salesman problem model

As illustrated in Figure 2, the graph enables the construction of candidate routes from different sequences of the 12 tourist destinations. The ACO algorithm evaluates these alternative sequences based on their total edge weights. A route with a smaller total weight represents a shorter travel distance and receives greater pheromone reinforcement during the route-search process. In this study, W1 was defined as the initial and final node, while W2 to W12 were visited once according to the sequence generated by each ant.

B. Experimental Design and Ant Colony Optimization Algorithm Mechanism

This study employs the Ant Colony Optimization algorithm as a metaheuristic method to solve the Traveling Salesman Problem on a weighted graph. Ant Colony Optimization is selected for its ability to handle large solution spaces through probabilistic exploration and reinforcement of promising paths via pheromone trails. This approach is inspired by the collective behavior of ant colonies in discovering the shortest paths to food sources through indirect communication mediated by pheromones.

All algorithm parameters in this study are defined as fixed parameters. The number of nodes is set to $n = 12$ according to the number of tourist destinations, while the number of ants is set to $m = 12$. The initial pheromone value on all graph edges is initialized to $\tau 0 = 0.01$. The pheromone influence parameter and heuristic influence parameters $\alpha=1$ and $\beta=1$ were used to apply equal exponents to pheromone intensity and heuristic information in the transition-weight calculation. to route selection. The pheromone constant is set to $Q = 1$. Route selection is performed stochastically using the Roulette Wheel method, while pheromone updating follows a best-only scheme, in which only the routes belonging to the best solution in each iteration receive pheromone reinforcement. Experimental variation is conducted on the pheromone evaporation rate parameter with values of $\rho = 0.3$, 0.5, and 0.7. All ants start their tours from the initial node W1 and construct closed routes by returning to node W1 at the end of each iteration.

The Ant Colony Optimization algorithm mechanism in this study consists of several main stages, namely parameter initialization, solution construction, pheromone updating, and termination evaluation. During the initialization stage, the system sets the initial pheromone values and algorithm parameters. In the solution construction stage, each ant incrementally builds a route by selecting the next node based on transition probabilities influenced by pheromone intensity and heuristic information. The heuristic value is determined as the inverse of the distance between nodes, such that paths with shorter distances have a higher probability of being selected (Armond et al., 2022; Blum, 2024). This process continues until all nodes have been visited and the ant returns to the starting node. The fundamental behavior underlying this mechanism is illustrated in Figure 3.

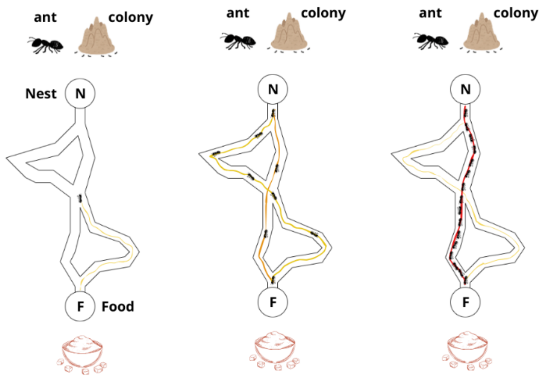

Figure 3. The behavior of ants

After all ants have completed their routes, the system calculates the total length of each route and determines the best route based on the minimum total distance. The pheromone updating stage is then performed by applying pheromone evaporation to all graph edges, followed by pheromone reinforcement only on the edges that belong to the best route. This process is repeated until the predefined maximum number of iterations is reached (Wang & Han, 2021).

C. Mathematical Formulation and Simulation Procedure

The transition probability of an ant moving from node i to node j is calculated based on the pheromone intensity τ_{ij} and the heuristic value η_{ij}, which is defined as the inverse of the distance between nodes, namely $1/d_{ij}$. The transition probability is expressed as a probabilistic function that incorporates the parameters α and β to control the relative influence of pheromone intensity and heuristic information. This formulation allows each ant to select its path stochastically, with a higher tendency to choose paths that have higher pheromone levels and shorter distances.

After all ants have completed their routes, the system calculates the length of each route, denoted as Lk. Pheromone updating is performed in two stages, namely evaporation and pheromone deposition. Pheromone evaporation is applied by multiplying the existing pheromone value by the factor (1 − ρ) to prevent the dominance of specific paths and to maintain diversity. Pheromone deposition is applied only to the paths that belong to the best route, with the pheromone increment Δτ calculated based on the ratio of Q to the length of the best route. As a result, routes with shorter total distances receive stronger reinforcement. This best-only updating scheme is intended to accelerate the algorithm's convergence toward an optimal solution.

The simulation procedure is performed iteratively for each value of the pheromone evaporation rate. In each iteration, all ants construct their routes stochastically using the Roulette Wheel mechanism. The system then calculates the route length of each ant and selects the best route based on the minimum Lk value. Paths that belong to the best route receive pheromone reinforcement, while the remaining paths undergo pheromone evaporation only. The thesis report includes an example of a manual calculation in the first iteration for a single ant to illustrate the path selection mechanism and transition probability computation in detail. This simulation procedure is used to evaluate the consistency of the algorithm in finding the minimum Hamiltonian cycle and to analyze the effect of varying pheromone evaporation rates on the quality of the resulting tourism route solutions.

For example, in the initial step from node W1, the candidate destinations range from W2 to W12. The base value used to calculate the transition probability is:

$$(Wn) = \frac{0{,}01}{dij} \tag{1}$$

This study also evaluates the effect of variations in the pheromone evaporation rate on the updated pheromone values. Pheromone updating is performed using the equation:

$$\tau_{ij} = (1 - \rho)\tau_{ij} + \sum_{k=1}^{m} \Delta\tau_{ij}^{k} \tag{2}$$

D. ACO Implementation Procedure

The implementation of the Ant Colony Optimization algorithm begins with parameter initialization and the placement of artificial ants at the starting point of the tour. Each ant then incrementally constructs a route by selecting the next destination based on probabilities influenced by pheromone values and inter-destination distances. The path selection process continues until all destinations have been visited and a complete route is formed.

After all ants have completed their respective routes, the solution quality is evaluated based on the resulting total travel distance. The pheromone values on each path are then updated, where routes with shorter distances receive stronger pheromone reinforcement, while other paths experience pheromone reduction due to the evaporation process. This mechanism enables the algorithm to iteratively emphasize better solutions. Figure 4 presents the implementation flow of the ACO algorithm for tourism route optimization

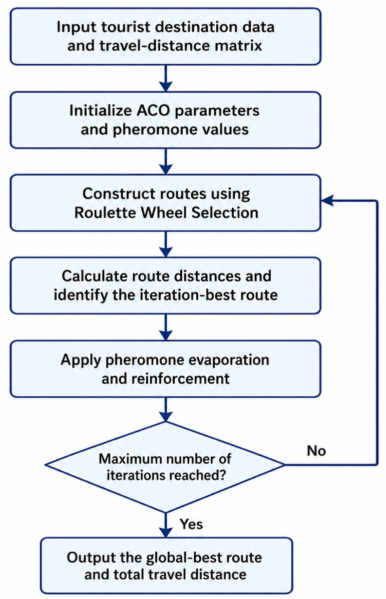

Figure 4. Flowchart of the ACO implementation

The process of solution construction and pheromone updating is repeated until the maximum number of iterations is reached or the termination condition is satisfied. The best solution obtained throughout all iterations is then determined as the optimal tourism route. This approach enables the Ant Colony Optimization algorithm to generate efficient routing solutions through a collective and iterative learning process.

RESULT AND DISCUSSION

This study uses distance data between tourist destinations obtained through Google Maps observations, considering the fastest road routes, as shown in Table 1.

Table 1. Tourist Destination Data

Symbol	Tourist Attraction Name	Latitude	Longitude
W1	Stasiun Purwokerto	7.419020156	109.2218972
W2	Alun-Alun Purwokerto	7.424100459	109.2301078
W3	Menara Pandang Teratai	7.431264851	109.2325642
W4	Taman Mas Kemambang	7.411501001	109.2372995
W5	The Village Purwokerto	7.374241978	109.2404834
W6	Lokawisata Baturraden	7.313213822	109.2290342
W7	Alun-Alun Purbalingga	7.389101885	109.363246
W8	Telaga Situ Tirta Marta	-7.33355394	109.3138169
W9	Taman Sanggaluri Park	7.356500738	109.3312153
W10	Owabong Waterpark	7.349399929	109.3497249
W11	Masjid Cheng Hoo Purbalingga	7.31748304	109.3620948
W12	D'Las Lembah Asri Serang	7.242232273	109.2911107

The use of Google Maps allows the researcher to obtain distance data that represent actual geographic conditions and road networks. These distance data are deterministic in nature and are used as static data throughout the simulation process. The inter-location distances serve as edge weights in the graph and directly influence the computation of heuristic values in the Ant Colony Optimization algorithm.

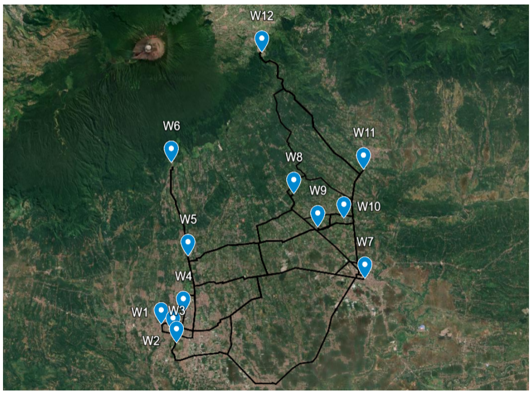

Figure 5. Map of Tourist Location Distribution

All distance data obtained are then organized into a weighted distance matrix to facilitate graph modeling and algorithm computation. This distance matrix represents the distances between each pair of tourist destinations in kilometers. Figure 5 presents each row and column corresponds to nodes W1 to W12, while the main diagonal values are zero, indicating the distance of a location to itself. The distance matrix is symmetric because the distance from node Wi to Wj is equal to the distance from node Wj to Wi. This distance matrix serves as the basis for constructing a complete weighted graph and is used to compute heuristic values in the Ant Colony Optimization algorithm, defined as the inverse of the inter-node distance, shown in Table 2.

The weighted graph represents the Traveling Salesman Problem structure, in which each generated closed route visits every node exactly once and returns to W1.

Table 2. Inter-Node Distance Data

Inter-Node Distance Matrix												
	W1	W2	W3	W4	W5	W6	W7	W8	W9	W10	W11	W12
W1	0	1,3	2,4	2,7	8,0	14,9	18,5	19,3	18,5	21,2	25,5	38,0
W2	1,3	0	1.1	2,4	7,7	14,6	18,2	19,0	18,2	21,2	25,1	37,7
W3	2,4	1.1	0	3,5	7,8	14,7	21,2	21,8	21,1	24,0	29,0	41,5
W4	2,7	2,4	3,5	0	4,8	11,7	16,1	16,9	16,1	19,0	22,8	35,6
W5	8,0	7,7	7,8	4,8	0	6,9	15,7	12,0	11,8	14,2	18,0	30,5
W6	14,9	14,6	14,7	11,7	6,9	0	22,6	18,9	18,2	21,1	24,9	37,4
W7	18,5	18,2	21,2	16,1	15,7	22,6	0	9,3	5,8	6,0	8,7	21,3
W8	19,3	19,0	21,8	16,9	12,0	18,9	9,3	0	4,2	6,0	9,5	15,2
W9	18,5	18,2	21,1	16,1	11,8	18,2	5,8	4,2	0	3,3	7,0	19,6
W10	21,2	21,2	24,0	19,0	14,2	21,1	6,0	6,0	3,3	0	4,5	17,1
W11	25,5	25,1	29,0	22,8	18,0	24,9	8,7	9,5	7,0	4,5	0	13,4
W12	38,0	37,7	41,5	35,6	30,5	37,4	21,3	15,2	19,6	17,1	13,4	0

The computation process of the Ant Colony Optimization algorithm begins with the initialization of the pheromone value to 0.01 on all graph edges. In the first iteration, all ants start their tours from node W1 and construct routes stochastically using the Roulette Wheel mechanism. The selection of the next node is influenced by the uniformly initialized pheromone values and the heuristic information derived from the distance matrix. At this stage, the influence of distance becomes dominant because the pheromone values have not yet been updated, as shown in Table 3.

These values are then summed to obtain the total probability weight, which is subsequently used to compute the probability of each node being selected as the next destination.

Table 3. Tabu List W1

Destination	Distance (dij)	0,01/dij
W2	1,3	0,007692
W3	2,4	0,004167
W4	2,7	0,003704
W5	8,0	0,001250
W6	14,9	0,000671
W7	18,5	0,000541
W8	19,3	0,000518
W9	18,5	0,000541
W10	21,2	0,000472
W11	25,5	0,000392
W12	38,0	0,000263

The transition probability values are then obtained by dividing each $0{,}01/d_{ij}$ value by the corresponding total sum. The calculation results indicate that nodes with shorter distances have higher probabilities of being selected.

The selection of the next node is performed using the Roulette Wheel Selection method, in which nodes with higher probabilities have a greater chance of being selected. This mechanism ensures that the search process remains stochastic while still being guided by distance-based information, as shown in Table 4.

Table 4. Probability W1

Destination	Probability
W2	0,3806
W3	0,2062
W4	0,1833
W5	0,0619
W6	0,0332
W7	0,0268
W8	0,0256
other	< 0,03

Although the transition probability from a node indicates a tendency to select the path with the highest value, the Ant Colony Optimization algorithm does not always choose the path with the maximum probability at each step. During the solution construction stage, the selection of the next node is performed stochastically using the Roulette Wheel mechanism, allowing paths with lower probabilities to still be selected. This condition reflects the exploratory nature of the Ant Colony Optimization algorithm, which aims to avoid premature convergence to local optima.

In one of the ant trajectories, a transition occurs from node W9 to W7, even though W7 does not have the highest transition probability among the candidates. This selection remains algorithmically valid because the probability value of W7 still lies within the probability distribution range and is chosen through random sampling in the roulette wheel process. This mechanism allows the algorithm to explore alternative paths that may lead to routes with shorter total distances in subsequent iterations. Therefore, the differences in ant trajectories during the early iterations are a direct consequence of the balance between exploration and exploitation, which is a fundamental characteristic of the Ant Colony Optimization algorithm, as shown in Table 5.

Table 5. Exploration from Node W9

Destination from W9	dij	0,01/dij	Pij	Cumulative Probability
W7	5,8	0,001724138	0,2415	0,2415
W8	4,2	0,002380952	0,3337	0,5752
W10	3,3	0,003030303	0,4247	0,9999

After all ants have completed their routes in the first iteration, the system calculates each ant's route length and selects the route with the minimum total distance. The results of the first iteration for the twelve ants are presented in Table 6 below.

Table 6. First Iteration Results of 12 ACO Ants

Ant	Route (looping)
K1	W1 - W4 - W2 - W3 - W5 - W10 - W9 - W11-W7 - W8 - W12 - W6 - W1
K2	W1 - W3 - W2 - W4 - W8 - W9 - W7 - W10 - W11 - W12 - W6 - W5 - W1
K3	W1 - W2 - W3 - W11 - W8 - W5 - W4 - W10 - W9 - W7 - W12 - W6 - W1
K4	W1 - W11 - W8 - W9 - W10 - W7 - W12 - W6 - W5 - W4 - W2 - W3 - W1
K5	W1 - W2 - W3 - W4 - W5 - W9 - W8 - W10 - W11 - W12 - W6 - W7 - W1
K6	W1 - W4 - W2 - W5 - W3 - W8 - W12 - W10 - W11 - W9 - W7 - W6 - W1
K7	W1 - W2 - W3 - W4 - W5 - W6 - W12 - W11 - W9 - W10 - W8 - W7 - W1
K8	W1 - W3 - W2 - W4 - W7 - W8 - W10 - W11 - W12 - W6 - W5 - W9 - W1
K9	W1 - W2 - W3 - W4 - W8 - W11 - W9 - W5 - W10 - W12 - W6 - W7 - W1
K10	W1 - W6 - W2 - W4 - W7 - W9 - W8 - W11 - W12 - W10 - W5 - W3 - W1
K11	W1 - W7 - W6 - W5 - W3 - W9 - W11 - W8 - W10 - W12 - W4 - W2 - W1
K12	W1 - W3 - W5 - W6 - W2 - W4 - W7 - W12 - W11 - W9 - W8 - W10 - W1

The criterion for selecting the best ant was the shortest total route length in the first iteration. Recalculation using the weighted distance matrix showed that K2 produced the shortest route, W1–W3–W2–W4–W8–W9–W7–W10–W11–W12–W6–W5–W1, with a total distance of 109.0 km. Therefore, K2 was designated as the best ant for the first pheromone update, with Δτbest = 1/109.0 = 0.009174, as shown in Table 7.

Table 7. Total Distance of the First Iteration

Ant	Total Distance (km)	Δτ
K1	124.0	0.008065
K2	109.0	0.009174
K3	159.4	0,006274
K4	124.8	0.008013
K5	129.1	0.007746
K6	129.5	0.007722
K7	112.5	0.008889
K8	129.8	0.007704
K9	160.9	0.006215
K10	122.4	0.008170
K11	155.8	0.006418
K12	123.3	0.008110

This route is used as the basis for pheromone updating using the best-only scheme, in which the graph edges that belong to the best route receive pheromone reinforcement, while the other edges undergo only pheromone evaporation. This mechanism strengthens the paths that produce optimal solutions and gradually guides the algorithm's search toward routes with minimum total distance.

The first update calculation shows that the pheromone evaporation rate ρ affected the amount of pheromone retained on graph edges. The shortest-route edges retained pheromone values of 0.016174, 0.014174, and 0.012174 for ρ values of 0.3, 0.5, and 0.7, respectively. A lower evaporation rate resulted in more pheromone being retained after this update, as shown in Table 8.

These values describe pheromone retention after the first update and do not independently demonstrate differences in final route quality, exploration, or convergence. Evaluating those effects requires repeated runs across multiple iterations for every evaporation-rate setting.

Table 8. Pheromone Variation

Edge	τ_new (ρ=0,3)	τ_new (ρ=0,5)	τ_new (ρ=0,7)
W1–W2 (best)	0.016174	0.014174	0.012174
W2–W3 (best)	0.016174	0.014174	0.012174
W3–W4 (best)	0.016174	0.014174	0.012174
W4–W5 (best)	0.016174	0.014174	0.012174
W5–W6 (best)	0.016174	0.014174	0.012174
W6–W12 (best)	0.016174	0.014174	0.012174
W12–W11 (best)	0.016174	0.014174	0.012174
W11–W9 (best)	0.016174	0.014174	0.012174
W9–W10 (best)	0.016174	0.014174	0.012174
W10–W8 (best)	0.016174	0.014174	0.012174
W8–W7 (best)	0.016174	0.014174	0.012174
W7–W1 (best)	0.016174	0.014174	0.012174
Edge lain (non-best)	0.007000	0.005000	0.003000

Overall, the road-distance data obtained from Google Maps were represented as a weighted distance matrix and processed sequentially by the ACO algorithm. Based on the routes reported for the first iteration, the algorithm identified K2 as the shortest route, with a total distance of 109.0 km.

CONCLUSION

This study implemented the Ant Colony Optimization algorithm to generate a tourism route formulated as a Traveling Salesman Problem. Road-distance data obtained from Google Maps were organized into a weighted distance matrix representing the connections among 12 tourist destinations in Banyumas and Purbalingga. The matrix was subsequently used by the ACO algorithm to calculate heuristic information, construct candidate routes, evaluate their total distances, and update pheromone values.

The shortest route obtained in the first iteration was W1–W3–W2–W4–W8–W9–W7–W10–W11–W12–W6–W5–W1, with a total distance of 109.0 km. The algorithm produced a closed route that visited each destination once and returned to the starting point. The evaluated evaporation rates of 0.3, 0.5, and 0.7 resulted in shortest-route pheromone values of 0.016174, 0.014174, and 0.012174, respectively, after the pheromone update. These results indicate that the evaporation rate affected the amount of pheromone retained during the updating process. However, the experiment did not verify global optimality or compare ACO with alternative TSP methods. Therefore, the identified route should be interpreted as the shortest route obtained under the experimental configuration. Future research should include repeated trials, ablation analysis, convergence evaluation, and comparisons with exact or alternative methods.

REFERENCES

Anggraeni, D. A. F., Dianutami, V. R., & Tyasnurita, R. (2024). Investigation of Simulated Annealing and Ant Colony optimization to Solve Delivery Routing Problem in Surabaya, Indonesia. *Procedia Computer Science*, *234*, 592–601. https://doi.org/10.1016/j.procs.2024.03.044

Armond, A. M., Prasetyo, Y. D., & Ediningrum, W. (2022). Application of Ant Colony Optimization (ACO) Algorithm to Optimize Trans Banyumas Bus Routes. *Proceedings - 2022 IEEE International Conference on Cybernetics and Computational Intelligence, CyberneticsCom 2022*, 132–137. https://doi.org/10.1109/CyberneticsCom55287.2022.9865394

Blum, C. (2024). Ant colony optimization: A bibliometric review. In *Physics of Life Reviews* (Vol. 51, pp. 87–95). Elsevier B.V. https://doi.org/10.1016/j.plrev.2024.09.014

Bondy, J. A., & Murty, U. S. R. (1976). *GRAPH THEORY WITH APPLICATIONS NORfH-HOLLAND New York • Amsterdam • Oxford*.

Buinoschi-Tirpescu, A., & Breaban, M. E. (2024). Solving Min-Max MTSP in a Reinforcement Learning context. *Procedia Computer Science*, *246*(C), 1001–1010. https://doi.org/10.1016/j.procs.2024.09.519

Chauhan, A. (2024). A 1.5-Approximation for Symmetric Euclidean Open Loop TSP. *IEEE Access*, *12*, 144509–144518. https://doi.org/10.1109/ACCESS.2024.3472282

Goel, M., Singh, J., & Kumar Bairwa, A. (2025). Performance Evaluation of Heuristic and

Meta-Heuristic Algorithms on Large-Scale Travelling Salesman Problem Instances. *IEEE Access*, *13*, 156988–157010. https://doi.org/10.1109/ACCESS.2025.3606531

Han, M., Du, Z., Yuen, K. F., Zhu, H., Li, Y., & Yuan, Q. (2024). Walrus optimizer: A novel nature-inspired metaheuristic algorithm. *Expert Systems with Applications*, *239*. https://doi.org/10.1016/j.eswa.2023.122413

Hjeij, M., & Vilks, A. (2023). A brief history of heuristics: how did research on heuristics evolve? In *Humanities and Social Sciences Communications* (Vol. 10, Issue 1). Springer Nature. https://doi.org/10.1057/s41599-023-01542-z

Linganathan, S., & Singamsetty, P. (2024). Genetic algorithm to the bi-objective multiple travelling salesman problem. *Alexandria Engineering Journal*, *90*, 98–111. https://doi.org/10.1016/j.aej.2024.01.048

M. Almufti, S., Ahmad Shaban, A., Arif Ali, Z., Ismael Ali, R., & A. Dela Fuente, J. (2023). Overview of Metaheuristic Algorithms. *Polaris Global Journal of Scholarly Research and Trends*, *2*(2), 10–32. https://doi.org/10.58429/pgjsrt.v2n2a144

Nourmohammadzadeh, A., & Voß, S. (2026). A matheuristic approach for the robust coloured travelling salesman problem with multiple depots. *European Journal of Operational Research*, *328*(2), 390–406. https://doi.org/10.1016/j.ejor.2025.06.018

Ochelska-Mierzejewska, J., Poniszewska-Maranda, A., & Marana, W. (2021). Selected genetic algorithms for vehicle routing problem solving. *Electronics (Switzerland)*, *10*(24). https://doi.org/10.3390/electronics10243147

Shi, Y., & Zhang, Y. (2021). The neural network methods for solving Traveling Salesman Problem. *Procedia Computer Science*, *199*, 681–686. https://doi.org/10.1016/j.procs.2022.01.084

Stützle, T., & Dorigo, M. (2004). *Ant Colony Optimization*. https://www.researchgate.net/publication/36146886

Tao, Q., Zhang, T., & Han, J. (2023). An Approximate Parallel Annealing Ising Machine for Solving Traveling Salesman Problems. *IEEE Embedded Systems Letters*, *15*(4), 226–229. https://doi.org/10.1109/LES.2023.3298739

Wang, Y., & Han, Z. (2021). Ant colony optimization for traveling salesman problem based on parameters optimization. *Applied Soft Computing*, *107*. https://doi.org/10.1016/j.asoc.2021.107439

Wangying, X., & Naiming, X. (2025). Scheduling and route planning for forests rescue: Applications with a novel ant colony optimization algorithm. *Engineering Applications of Artificial Intelligence*, *155*. https://doi.org/10.1016/j.engappai.2025.111042

Zeng, X., Song, Q., Yao, S., Tian, Z., & Liu, Q. (2021). Traveling Salesman Problems with Replenishment Arcs and Improved Ant Colony Algorithms. *IEEE Access*, *9*, 101042–101051. https://doi.org/10.1109/ACCESS.2021.3093295