

Article

Optimizing Electric Delivery Vehicle Route Planning: A Hybrid Approach Integrating Clustering and Ant Colony Algorithm for Sustainable Transportation

Si Yong Heng , Anurag Sharma *  and Jianfang Xiao 

Electric Power Engineering, Newcastle University in Singapore, 1 Punggol Coast Road, Level 2 SIT Campus, Block E1, Singapore 828608, Singapore; s.y.heng2@newcastle.ac.uk (S.Y.H.); jianfang.xiao@newcastle.ac.uk (J.X.)

* Correspondence: anurag.sharma@newcastle.ac.uk

Abstract

The transition to electric vehicles (EVs) in urban logistics presents complex operational challenges, driven primarily by limited battery capacities, charging station scheduling, and dynamic traffic congestion. This paper introduces a framework to solve the Capacitated Multi-Depot Electric Vehicle Routing Problem (MD-EVRP). We propose a novel Multi-Depot Rotational Sweep Cluster K-means (MD-RSCK) algorithm to partition large-scale spatial data while strictly adhering to vehicle capacity constraints. To optimize intra-cluster routing, we develop an Ant Colony Optimization (ACO) engine augmented with a Time-Dependent Congestion Model. Furthermore, the framework integrates an Energy-Aware Route Refiner (EARR). This architecture utilizes recursive backtracking to ensure battery-feasible routes, adapting to both symmetric Euclidean approximations and real-world asymmetric traffic networks. The framework is evaluated against standard IEEE EVRP benchmarks and a multi-depot urban case study based on the road network of Shanghai, China. Experimental results demonstrate that this integrated architecture achieves competitive distance and cost metrics within a 2.44% optimality gap of state-of-the-art algorithms while ensuring strictly feasible battery states and preventing cyclic entrapment, providing a scalable operational tool for modern sustainable logistics.

Keywords: electric vehicle routing problem; ant colony optimization; K-means clustering; sustainable urban logistics; metaheuristics

1. Introduction

Electric delivery vehicles (EDVs) are rapidly transforming urban logistics, offering a sustainable alternative to conventional combustion-engine fleets [1]. However, their integration introduces significant operational challenges due to limited battery capacities, sparse charging infrastructure, and unpredictable traffic congestion. The Electric Vehicle Routing Problem (EVRP), an extension of the classical Vehicle Routing Problem (VRP), explicitly incorporates these battery constraints and charging objectives, aiming to minimize travel distance and energy consumption while scheduling charging stops to balance route duration with vehicle autonomy [2]. In real-world urban delivery operations, customer time windows specifying preferred or mandatory service intervals add further logistical complexity, giving rise to the Electric Vehicle Routing Problem with Time Windows (EVRPTW) [3]. The EVRPTW requires feasible routes that respect both range limitations and temporal constraints, often amid dynamic traffic that critically affects travel times and energy use.



Academic Editor: Luca D'Acierno

Received: 26 May 2026

Revised: 21 June 2026

Accepted: 26 June 2026

Published: 1 July 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and

conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

While conventional approaches like Mixed-Integer Linear Programming (MILP) are solvable for small instances, their computational complexity becomes prohibitive as network size increases, severely limiting their scalability for daily urban logistics [4]. Given these challenges, metaheuristic frameworks have emerged as effective alternatives for exploring the complex EVRPTW solution space [5]. Among candidates, genetic algorithms (GAs) offer strong global search capabilities but can suffer from slow convergence in large networks and sensitivity to operators that disrupt spatial structures [6]. Particle swarm optimization (PSO) excels in continuous optimization but is less suited to discrete routing with hard constraints, often prone to premature convergence [7]. Simulated annealing (SA) [8], Tabu search (TS) [9], and variable neighborhood search (VNS) [10] provide effective local search escapes but can demand excessive calculation times in high-dimensional problems, often requiring enhancements like memory-based restarts to balance exploration and exploitation.

Among these metaheuristics, Ant Colony Optimization (ACO) stands out for routing due to its proven strengths in combinatorial problems [11]. Its constructive, stigmergic mechanism adapts to sequential path dependencies, driving developments in energy-aware ACO variants [12]. However, a review of the recent literature reveals structural gaps in how these algorithms are applied to the modern EVRPTW. Much of the existing EVRP literature prioritizes single-objective optimization, typically minimizing distance or energy consumption [13]. It provides limited explicit balancing of time, cost, and battery factors, as exemplified by collaborative models utilizing fixed-weight recharging structures [14,15]. Additionally, many models rely on static routing assumptions and neglect dynamic factors such as time-dependent traffic congestion profiles that govern practical route feasibility [16].

To address these baseline limitations, recent studies have introduced adaptive modifications to standard pheromone update rules. For instance, Zhang et al. [17] employed a clone adaptive ACO approach to balance search exploration and exploitation while minimizing economic and temporal costs in express delivery routing. Extending adaptive mechanisms to electric mobility, Tan et al. [18] utilized an improved ACO engine that integrates dynamic charging station selection and discrete energy constraints directly into the path-finding process.

While these models advance intra-route optimization, they exhibit structural limitations when applied to multi-vehicle logistics. The model proposed by Zhang et al. operates without energy constraints, whereas the charging framework by Tan et al. evaluates single-vehicle itineraries. The CACO architecture proposed in this paper integrates these isolated mechanics into a unified solution. By combining a stagnation-resilient MAX–MIN Ant System (MMAS) routing engine with a multi-depot spatial clustering algorithm (MD-RSCK), the framework scales the adaptive search principles and embedded charging logic across multi-vehicle fleets distributed over large-scale urban networks. This hierarchical structure ensures system-wide payload compliance prior to individual route optimization, extending the single-vehicle models into a mathematically feasible multi-depot architecture.

First, while state-of-the-art algorithms have shown promise on standard benchmarks, they typically employ monolithic search strategies that scale poorly as network size increases. For instance, the Adaptive Large Neighborhood Search (ALNS) proposed by Wang et al. [19] and the bilevel ACO implementations developed by Jia et al. [20,21] demonstrate strong theoretical performance but lack the spatial decomposition necessary for large-scale urban fleets. When spatial decomposition is attempted (e.g., hybrid ACO [22] or spatial-temporal clustering [23]), it heavily relies on isotropic methods such as standard K-means or density-based clustering. These methods possess an inherent “compactness bias”, forming circular delivery zones that severely misalign with the anisotropic, elongated nature of urban road networks, inducing excessive cross-corridor detours [24].

Second, current EVRP methodologies struggle with the efficient handling of battery capacity constraints. Jia et al. [20,21] treated the EVRP as a bilevel problem, generating energy-unaware customer routes at an upper level and attempting to schedule charging at a lower level using simple enumeration or fixed heuristics. Other recent models similarly defer strict electricity enforcement by relying on soft-penalty relaxations. For instance, Thymianis et al. [25] utilized penalty functions to temporarily manage battery depletion during the initial heuristic route construction. To address unpredictable transit conditions, Jeong et al. [26] incorporated penalty-based relaxations to handle energy consumption uncertainty within their adaptive robust routing models. Furthermore, Wang et al. [27] extended this penalty reliance to multi-compartment logistics, using heuristic adjustments to balance battery drain against strict temperature and humidity settings for perishable deliveries. By deferring charging decisions to post-construction repair phases or relying on penalties, these methods allow mathematically infeasible candidate solutions to persist during the search. This results in wasted computational effort, delayed convergence, and the risk of stranded vehicles in practical deployments. Furthermore, these algorithms almost exclusively assume symmetric Euclidean distance matrices, making them highly susceptible to cyclic entrapment (e.g., infinite loops) when applied to the asymmetric, directed graphs of urban traffic.

While the studies discussed above focus primarily on single-depot EVRP configurations, real-world urban logistics frequently operate from multiple depots distributed across a city. Recognizing this gap, recent research has begun to address the Multi-Depot Electric Vehicle Routing Problem (MD-EVRP) and its variants. For instance, Fan [22] proposed a two-stage hybrid ant colony algorithm for the Multi-Depot Half-Open Time-Dependent Electric Vehicle Routing Problem (MDHOTDEVPR). The first stage uses enhanced K-means clustering to assign customers to EVs, while the second stage employs an Improved Ant Colony Algorithm (IACA) for intra-cluster route optimization. Similarly, Wang and Chen [28] formulated the MDEVPRPTW as a mixed-integer linear programming model and developed an adaptive large neighborhood search (ALNS) with problem-specific destroy and repair operators to solve it efficiently.

However, a closer examination of these MD-EVRP studies reveals persistent limitations. The clustering stage in Fan's approach [22] remains fundamentally capacity-unaware, relying on standard K-means which possesses an inherent 'compactness bias', forming circular delivery zones that misalign with the anisotropic nature of urban road networks. Meanwhile, the ALNS framework proposed by Wang and Chen [28] treats charging feasibility as a decoupled decoding step rather than embedding energy constraints directly into route construction. By optimizing route sequences on a giant-tour representation and inserting charging infrastructure retroactively during evaluation, this decoupled strategy allows mathematically infeasible paths to persist during local search moves, delaying convergence and risking stranded vehicles in practice. More broadly, the existing MD-EVRP literature continues to rely on isotropic clustering, penalty-based charging relaxations, and largely assumes symmetric distance matrices, leaving critical gaps in scalability and real-world applicability.

In summary, existing literature on the EVRP can be categorized into three methodological tiers. The first tier includes exact methods, which guarantee optimality but fail on realistically sized urban datasets due to combinatorial explosion. The second tier comprises global metaheuristics (e.g., standard GA, ACO, SA), which scale better but frequently violate strict battery physics when evaluated holistically. The third tier consists of decomposition frameworks (cluster-first, route-second models). However, current decomposition approaches predominantly utilize capacity-unaware clustering (e.g., standard K-means), severely limiting their applicability to the Multi-Depot EVRP (MD-EVRP). This paper iden-

tified several critical gaps in the existing literature: the scalability bottlenecks of monolithic searches, the isotropic bias of standard clustering, the inefficiencies of decoupled charging repairs, and the lack of asymmetric traffic adaptability. To address these gaps, the Clustering and Ant Colony Optimization (CACO) framework is proposed. To the best of our knowledge, this is the first integrated framework that combines deterministic capacitated sweep clustering with a proactive charging-aware ACO state transition and a recursive backtracking feasibility enforcer for the multi-depot EVRP.

The principal contributions of this work are as follows:

- **MD-RSCK Clustering Algorithm:** Integrates a radial sweep heuristic to generate deterministic, capacity-constrained initial centroids for K-means. This physically aligns delivery clusters with urban road networks while strictly enforcing maximum vehicle payload limits.
- **Proactive Recharging MMAS:** Embeds exact state-of-charge (SoC) look-aheads directly into the ant colony state-transition rule. This dynamically inserts charging events during route construction, preventing the generation of energy-unaware, mathematically infeasible solutions.
- **Energy-Aware Route Refiner (EARR):** Applies a load-dependent energy consumption model post-optimization. It uses recursive backtracking to sequence or prune charging events, ensuring strictly feasible battery trajectories before dispatch.
- **Empirical Validation:** Evaluates the framework against IEEE EVRP benchmarks and a multi-depot Shanghai logistics dataset. Results demonstrate highly competitive routing optimality alongside mathematically guaranteed adherence to physical payload capacities and strict battery constraints.

2. Problem Description and Formulation

The Electric Vehicle Routing Problem with Time Windows (EVRPTW) is formulated on a complete directed graph $G = (V, E)$. The node set $V = C \cup S \cup D$ comprises customer nodes $C = \{1, 2, \dots, n\}$, charging station nodes $S = \{1, 2, \dots, s\}$, and multiple depot nodes $D = \{d_1, d_2, \dots, d_p\}$. A homogeneous fleet of electric delivery vehicles (EDVs), denoted by $K = \{1, 2, \dots, m\}$, is distributed across the depots, with each vehicle required to start and end its route at its assigned depot.

2.1. Multi-Objective Function

The primary objective is to minimize the fleet-wide weighted combination of normalized travel time and operational cost. The global objective function F is defined in Equation (1) as the sum of the combined time-cost values (TC_k) for all vehicles $k \in K$:

$$F = \sum_{k \in K} TC_k \quad (1)$$

Equation (2) computes the individual combined time-cost value for each vehicle k :

$$TC_k = (1 - \omega)\bar{T}_k + \omega\bar{C}_k \quad (2)$$

Here, the weighting parameter $\omega \in [0, 1]$ balances the temporal and economic objectives, allowing fleet operators to dynamically adjust routing priorities. Because the total route travel time (T_k , measured in minutes) and the total operational route cost (C_k , measured in monetary units) operate on disparate scales, Equations (3) and (4) apply a dynamic min–max normalization to calculate the final fitness of a completed route. Rather than relying on static global bounds, the minimum (T_{min}) and maximum (T_{max}) travel times, alongside the minimum (C_{min}) and maximum (C_{max}) operational costs, were dynamically

extracted from the bounds of the candidate solutions evaluated within the current search space. This ensures that the framework accurately evaluates relative route performance dynamically as the optimization process converges:

$$\bar{T}_k = \frac{(T_k - T_{min})}{(T_{max} - T_{min})} \quad (3)$$

$$\bar{C}_k = \frac{(C_k - C_{min})}{(C_{max} - C_{min})} \quad (4)$$

Equation (5) calculates the total travel time T_k , which encompasses the driving time on all traversed arcs (T_{ijk}), the service time at customer locations ($t_{service,i}$), and the recharging duration at charging stations ($T_{charging,sk}$). Equation (6) computes the total cost C_k , comprising a fixed daily vehicle deployment cost (C_f), dynamic segment travel costs (C_{ijk}), charging expenses ($C_{charging,kl}$), and time-window violation penalties (C_{pij}):

$$T_k = \sum_{ij} x_{ijk} T_{ijk} + \sum_i \delta_{ik} \cdot t_{service,i} + \sum_z T_{charging,sk} \quad (5)$$

$$C_k = C_f + \sum_{ij} x_{ijk} C_{ijk} + \sum_l C_{charging,kl} + \sum_j y_{jk} C_{pij} \quad (6)$$

In these equations, x_{ijk} is a binary decision variable equal to 1 if vehicle k traverses the arc from node i to node j ; δ_{ik} is a binary indicator equal to 1 if vehicle k visits node i ; and y_{jk} is a binary indicator equal to 1 if customer i is assigned to vehicle k .

2.2. Routing and Resource Constraints

To ensure the structural and operational feasibility of the generated delivery routes, the model enforces a strict set of routing, capacity, and energy constraints.

Customer Fulfillment and Flow Conservation: Equation (7) ensures that every customer j is visited once by exactly one vehicle. Route continuity is preserved through Equation (8), which ensures flow conservation for all nodes entering and exiting a location:

$$\sum_{i \in V, i \neq j} \sum_{k \in K} x_{ijk} = 1, \forall j \in C \quad (7)$$

$$\sum_{i \in V \setminus \{j\}} x_{ijk} - \sum_{i \in V \setminus \{j\}} x_{jik} = 0, \forall j \in C \cup S, k \in K \quad (8)$$

Depot Assignment: Equation (9) enforces that each vehicle k must depart from and ultimately return to its designated depot d_k :

$$\sum_{j \in V \setminus \{D\}} x_{d_k,j,k} = 1, \sum_{i \in V \setminus \{D\}} x_{i,d_k,k} = 1 \quad \forall k \in K \quad (9)$$

Vehicle Payload Capacity: Equation (10) restricts the cumulative demand q_j of all customers served by vehicle k so that it cannot exceed the vehicle's maximum payload capacity Q_k :

$$\sum_{j \in C} q_j \left(\sum_{i \in V \setminus \{j\}} x_{ijk} \right) \leq Q_k, \forall k \in K \quad (10)$$

Battery State-of-Charge (SoC): To prevent battery depletion and preserve battery health, Equation (11) enforces that the state-of-charge for vehicle k arriving at any node i (SoC_{ik}) must remain above the minimum safety threshold (SoC_{min}) at all times:

$$SoC_{ik} \geq SoC_{min}, \forall i \in V, \forall k \in K \quad (11)$$

2.3. Auxiliary Operational Models

To capture the complex dynamics of urban logistics, the framework incorporates specific auxiliary models for traffic congestion, load-dependent energy consumption, and temporal penalties.

Time-Dependent Traffic Model: The travel time between nodes is calculated using a Bureau of Public Roads (BPR) type function [29] to model dynamic urban congestion. The free-flow time T_{ij}^0 is calculated in Equation (12) based on the segment distance d_{ij} and the maximum vehicle speed V_{max} . Equation (13) scales this time using a time-dependent congestion factor C_τ . Rather than relying on continuous live data feeds, the model extracts C_τ from established time-of-day congestion profiles (e.g., morning rush, mid-day, evening peak). Here, α_T and β_T represent calibrated road-network coefficients, and γ represents additional urban intersection delays:

$$T_{ij}^0 = \frac{d_{ij}}{V_{max}} \quad (12)$$

$$T_{ijk} = T_{ij}^0 \left(1 + \alpha_T \cdot \left(\frac{C_\tau}{100} \right)^{\beta_T} + \gamma \right) \quad (13)$$

Segment Cost Model: Equation (14) calculates the operational cost per segment (C_{ijk}). It incorporates a per-delivery reward (C_D) applied solely when traveling to a customer node via the indicator $I\{j \in C\}$ and the driver's hourly wage rate (λ) multiplied by the travel time:

$$C_{ijk} = C_D \cdot I\{j \in C\} + \lambda T_{ijk} \quad (14)$$

Charging Operations: Equations (15) and (16) model the charging operations. The time required to recharge ($T_{charging,sk}$) and the cost of the electricity ($C_{charging,sk}$) depend on the maximum battery capacity (E_{cap}), the station's specific charging power ($P_{charging,s}$), the electricity cost per kWh (e_c), and the difference between the vehicle's arrival SoC (SoC_i) and the target departure SoC (SoC_f):

$$T_{charging,sk} = \frac{E_{cap}(1 - SoC_{ik})}{P_{charging,s}} \quad (15)$$

$$C_{charging,sk} = (SoC_f - SoC_i) \cdot E_{cap} \cdot e_c \quad (16)$$

Load-Dependent Energy Consumption: Equation (17) computes the load-dependent energy consumed on an arc (E_{ijk}) dynamically based on the vehicle's current payload mass. Here, μ represents the base consumption rate, ν is the load-dependent energy factor, and $W_{d_k,k}$ is the initial payload departing the depot. The payload decreases progressively as deliveries (q_l) are fulfilled prior to reaching node j :

$$E_{ijk} = d_{ij} \left(\mu + \nu \cdot \left(W_{d_k,k} - \sum_{\substack{l \in C \\ y_{lk}=1 \\ l \text{ before } j}} q_l \right) \right) \quad (17)$$

Time Window Penalties: While vehicle capacity and battery limits are strictly enforced as hard constraints, time windows $[t_{E,i}, t_{L,i}]$ are modeled as soft constraints. Equation (18)

determines the penalty cost (C_{Pij}) applied at a penalty rate (C_{pv}) if the arrival time (t_{arr}) is outside the designated customer window:

$$C_{Pij} = \begin{cases} C_{pv} \cdot \max(t_{E,i} - t_{arr}, 0) & \text{if } t_{arr} < t_{E,i} \\ C_{pv} \cdot \max(t_{arr} - t_{L,i}, 0) & \text{if } t_{arr} > t_{L,i} \\ 0 & \text{otherwise} \end{cases} \quad (18)$$

3. Methodology: The CACO Framework

To efficiently solve the large-scale EVRPTW under multi-attribute constraints (vehicle capacity, strict battery limits, and time windows) and varying graph topologies, we propose the enhanced Clustering and Ant Colony Optimization (CACO) framework. CACO transitions from a monolithic metaheuristic approach to a structured, multi-stage pipeline designed to ensure 100% battery-feasible and capacity-compliant solutions while actively minimizing soft-penalty temporal violations. The overall architecture is executed in four sequential phases, as illustrated in Figure 1: (1) Data Processing and Graph Initialization; (2) Problem Decomposition via MD-RSCK; (3) Intra-Cluster Optimization via a Stagnation-Resilient MAX-MIN Ant System (MMAS); and (4) Deterministic Feasibility Assurance via Memetic Local Search (specifically, 2-Opt and 1-0 Shift operators) and Recursive Backtracking.

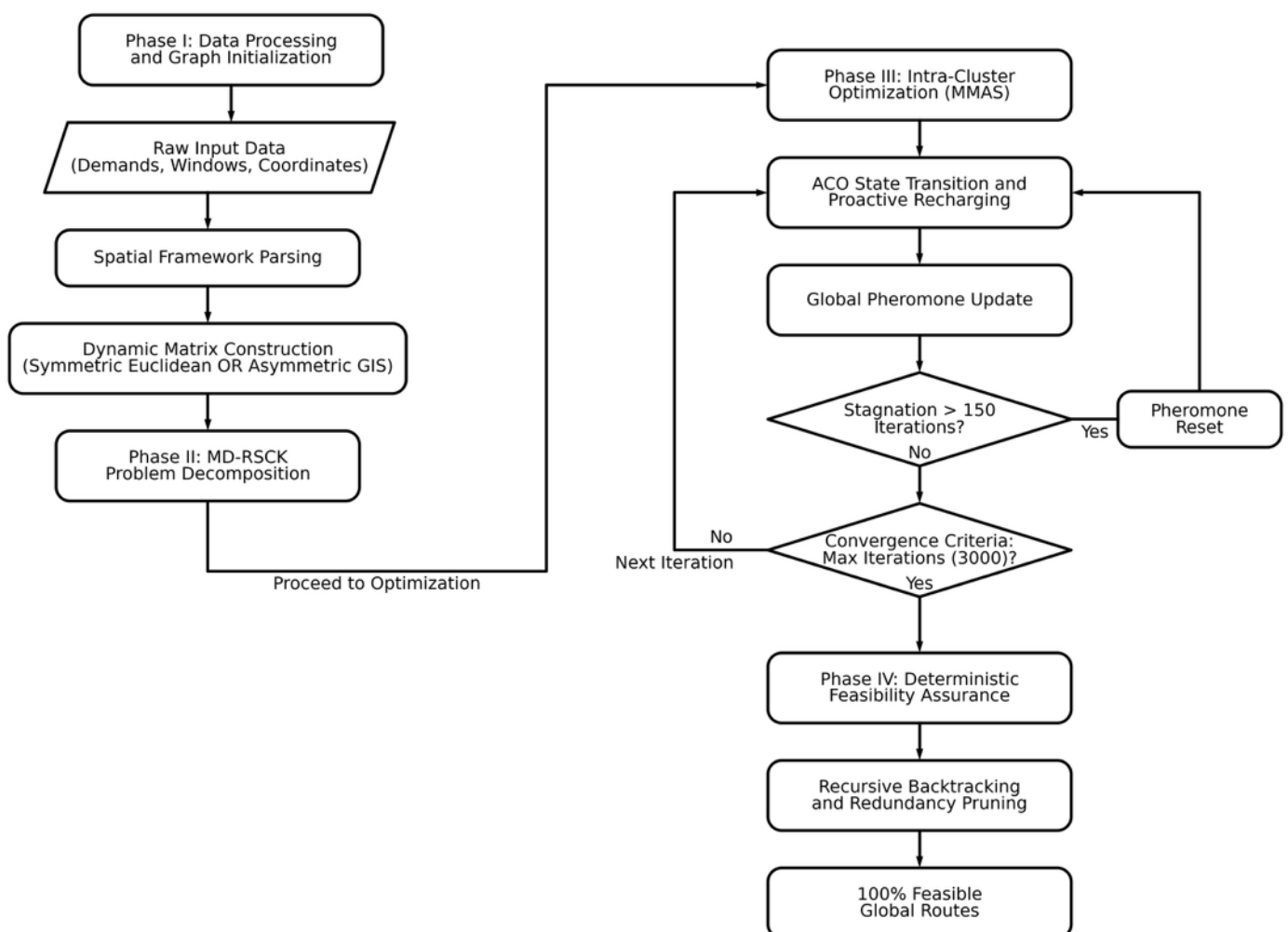


Figure 1. Optimization flowchart.

3.1. Phase I: Data Processing and Graph Initialization

The framework begins by establishing the spatial and operational foundation for the EDV fleet. Customer demands, time windows, depot locations, and charging station coordinates are ingested and parsed into a unified spatial framework. To bridge the gap between theoretical models and real-world deployment, the framework dynamically constructs the distance and time matrices. For standardized IEEE benchmarks, symmetric Euclidean distance matrices are computed. Conversely, for real-world urban environments, the framework interfaces directly with geospatial routing engines to generate highly asymmetric, directed network matrices, seamlessly applying the discrete profile-based traffic model previously defined in Equation (12).

3.2. Phase II: Problem Decomposition via Multi-Depot Rotational Sweep Cluster K-Means (MD-RSCK)

To ensure computational scalability without succumbing to the isotropic compactness bias of standard partitioned clustering, the global routing problem was decomposed using a Multi-Depot Rotational Sweep Cluster K-Means (MD-RSCK) algorithm. The step-by-step decomposition process, summarized in Algorithm 1, is executed as follows:

Algorithm 1 Multi-Depot Rotational Sweep Cluster K-Means (MD-RSCK)

Input: Customer set C , Depot set D , Vehicle capacity Q
Output: Feasible delivery clusters K

- 1: Assign each customer $i \in C$ to nearest depot $d \in D$
- 2: Initialize $\text{best_dispersion} = \infty$, $\text{best_seeds} = \emptyset$
- 3: **for** $\text{offset_angle} = 0^\circ$ TO 360° STEP 15° **do**
- 4: $\text{current_seeds} = \emptyset$, $\text{current_load} = 0$, $\text{current_wedge} = \emptyset$
- 5: Shift all customer polar angles by offset_angle (Equation (20))
- 6: Sort customers ascending by shifted angle θ'
- 7: **for** each customer i **DO**
- 8: **if** $\text{current_load} + \text{demand}[i] > Q$ **THEN**
- 9: Calculate geometric centroid (x^-, y^-) of current_wedge (Equations (21) and (22))
- 10: Append (x^-, y^-) to current_seeds
- 11: $\text{current_load} = 0$, $\text{current_wedge} = \emptyset$
- 12: **end if**
- 13: $\text{current_load} = \text{current_load} + \text{demand}[i]$
- 14: Add i to current_wedge
- 15: **end for**
- 16: Calculate Dispersion Score DS for current_seeds (Equation (23))
- 17: **if** $DS < \text{best_dispersion}$ **then**
- 18: $\text{best_dispersion} = DS$
- 19: $\text{best_seeds} = \text{current_seeds}$
- 20: **end if**
- 21: **end for**
- 22: Run K-Means($\text{init} = \text{best_seeds}$) on C
- 23: Execute Capacitated First-Fit Decreasing assignment
- 24: **return** final clusters K

Step 1: Global Depot Assignment: Customers are first assigned to their nearest active depot $d_k \in D$ using a centroid-proximity calculation, effectively partitioning the multi-depot problem into independent single-depot subproblems.

Step 2: Polar Coordinate Transformation: For a given depot located at (x_d, y_d) , the spatial coordinates (x_i, y_i) of each assigned customer i are transformed into a polar angle $\theta_i \in [-\pi, \pi]$ relative to the depot, as defined in Equation (19):

$$\theta_i = \arctan2(y_i - y_d, x_i - x_d) \quad (19)$$

To systematically test different geometric starting points, a rotational angular offset θ_{offset} is applied, mapping the shifted angle θ'_i to a standard $[0, 2\pi]$ range via Equation (20):

$$\theta'_i = (\theta_i - \theta_{offset}) \pmod{2\pi} \quad (20)$$

Step 3: Capacitated Radial Sweep Seeding: Customers are sorted in ascending order based on their shifted angle θ'_i . The algorithm sweeps circularly, accumulating customer demand q_i . Once the cumulative demand reaches the maximum vehicle payload capacity Q_k , a deterministic “capacity-constrained seed” is generated for the newly formed cluster c . This seed is located at the spatial centroid $(\bar{x}_c, \bar{y}_c)$, calculated as the arithmetic mean of the coordinates of all customers assigned to that wedge, as defined in Equations (21) and (22):

$$\bar{x}_c = \frac{1}{|c|} \sum_{i \in c} x_i \quad (21)$$

$$\bar{y}_c = \frac{1}{|c|} \sum_{i \in c} y_i \quad (22)$$

Here, $|c|$ represents the total number of customers within the formed wedge. Once the centroid seed is generated and stored, the demand counter is reset, and the sweep continues for the next vehicle.

Step 4: Dispersion Optimization: Because the starting angle heavily influences the final cluster shapes, the MD-RSCK algorithm systematically evaluates θ_{offset} in fixed 15° increments from 0° to 360° . For each iteration, it calculates a global dispersion score (DS) using Equation (23), where $\bar{x}_c$ and $\bar{y}_c$ represent the centroid coordinates of cluster c . The algorithm locks in the θ_{offset} that minimizes DS , ensuring the tightest spatial grouping:

$$DS = \sum_{c \in \text{Clusters}} \sum_{i \in c} \sqrt{(x_i - \bar{x}_c)^2 + (y_i - \bar{y}_c)^2} \quad (23)$$

To visually illustrate the mechanics of this dispersion optimization, Figure 2 demonstrates the rotational sweep profile applied to the E-n76-k7 benchmark instance. As the MD-RSCK algorithm systematically evaluates the 15° angular offsets, the resulting total dispersion score fluctuates significantly. The algorithm successfully identifies the optimal sweep angle, which significantly reduces spatial variance compared to the arbitrary 0° baseline typically utilized by standard K-means initialization.

Step 5: Deterministic K-Means Alignment and Final Capacitated Assignment: Traditional K-means clustering relies on randomized starting points (e.g., K-Means++), which are capacity-unaware and frequently generate overlapping or severely overloaded delivery zones. To solve this, the MD-RSCK algorithm bypasses random seeding entirely. Instead, the deterministic centroids $(\bar{x}_c, \bar{y}_c)$ calculated from the optimal radial sweep are injected directly into the K-means algorithm as fixed initial seeds. The algorithm executes in two final stages.

Geometric Smoothing: K-means runs its standard optimization loop, allowing the injected centroids to naturally shift toward the geographic density centers of the customer distribution. This smooths out the rigid, artificial borders of the initial radial wedges.

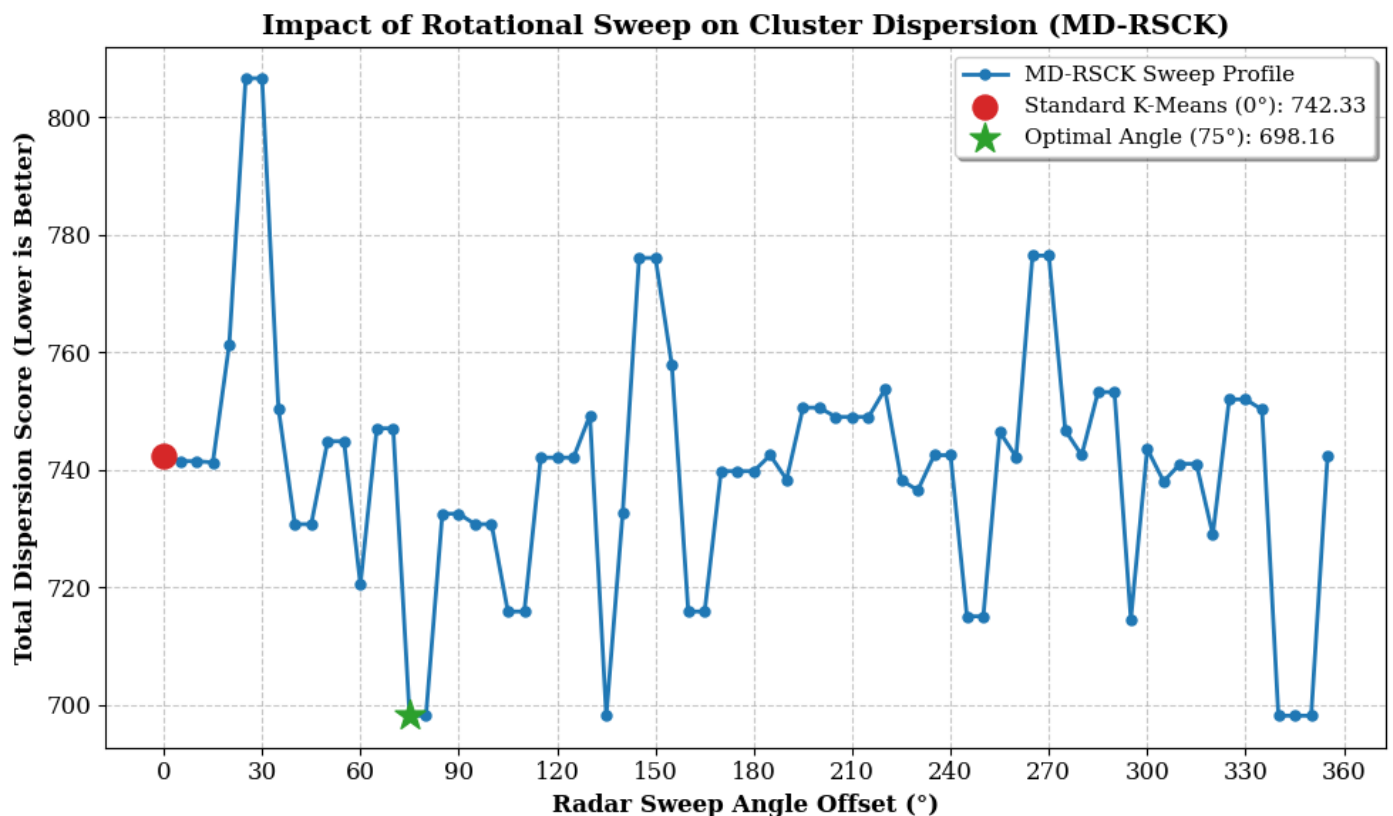


Figure 2. Impact of the rotational sweep offset angle on spatial dispersion. The MD-RSCK algorithm systematically evaluates angular offsets to minimize the total dispersion score. As shown, the optimal algorithmic sweep (75°) reduces spatial variance compared to the arbitrary 0° baseline utilized by standard K-means initialization.

Capacitated Node Assignment: To generate the definitive, mathematically feasible delivery clusters, a strict capacity-constrained assignment protocol is applied. Customers are sorted in descending order of demand and iteratively assigned to the nearest optimized centroid. If assigning a customer to their closest centroid would exceed the vehicle payload capacity (Q_k), they are assigned to the next-closest centroid with available capacity.

Furthermore, the MD-RSCK decomposition mathematically reduces the computational complexity of the subsequent routing phase. A standard global ACO evaluates an unconstrained search space with a complexity of roughly $O(n^2 \cdot m)$ per iteration, where n is the total number of nodes and m is the total number of ants. Because the CACO framework dynamically scales the swarm size to the dimensionality of the specific routing zone, partitioning the network into K distinct operational clusters distributes the workload, assigning roughly n/K nodes and m/K ants to each independent subproblem. The total sequential routing complexity across all clusters is therefore reduced to $K \cdot O((n/K)^2 \cdot (m/K))$, which mathematically simplifies to $O(n^2 m / K^2)$ per iteration. This quadratic reduction prevents combinatorial explosion, ensuring that the framework maintains the real-time execution speeds required for dynamic urban dispatching.

3.3. Phase III: Intra-Cluster Optimization via Stagnation-Resilient MMAS

Following problem decomposition, CACO performs independent route optimization within each cluster using a tailored memetic MMAS.

Step 1: Heuristic Desirability Calculation: As an ant evaluates moving from current node i to unvisited node j , the heuristic desirability η_{ij} is calculated using Equation (24). To balance the differing scales of segment travel time (t_{ij}) and segment operational cost (c_{ij}),

the variables are min–max normalized at the individual arc level into $\hat{t}_{ij}$ and $\hat{c}_{ij}$. To prevent unnecessary computational overhead during the metaheuristic execution, these boundary parameters ($t_{seg,min}$, $t_{seg,max}$, $c_{seg,min}$, $c_{seg,max}$) are treated as static constants. They are extracted exactly once from the isolated cluster's distance and traffic matrices immediately following the Phase II decomposition. They represent the absolute shortest/longest and cheapest/most expensive single-edge traversals available strictly within that specific spatial zone, ensuring that the normalization bounds remain stable across all ant iterations. The user-defined trade-off parameter $\rho_{tradeoff} \in [0, 1]$ controls the heuristic priority, and ε prevents zero-division:

$$\eta_{ij} = \frac{1}{(1 - \rho_{tradeoff})\hat{t}_{ij} + \rho_{tradeoff}\hat{c}_{ij} + \varepsilon} \quad (24)$$

Step 2: ACO State Transition: An ant at node i selects the next node j utilizing a stochastic roulette-wheel selection governed by the transition probability P_{ij} , defined in Equation (25). Here, τ_{ij} represents the current pheromone concentration on the edge, while α and β are parameters controlling the relative influence of the pheromone trail versus the heuristic visibility:

$$P_{ij} = \frac{\tau_{ij}^{\alpha} \cdot \eta_{ij}^{\beta}}{\sum_{l \in \text{unvisited}} \tau_{il}^{\alpha} \cdot \eta_{il}^{\beta}} \quad (25)$$

Step 3: Embedded Proactive Recharging: Rather than generating energy-unaware routes, the ACO incorporates a traffic-aware state-of-charge (SoC) look-ahead. Before completing the transition to j , the exact load-dependent energy consumption E_{ijk} is evaluated via Equation (17). If the projected SoC falls below the safety threshold SoC_{min} , the ant autonomously preempts the visit. It scans the charging infrastructure S and selects the station s^* that minimizes the temporal detour, taking dynamic traffic multipliers into account, as detailed in Algorithm 2.

Step 4: Pheromone Update and Stagnation Reset: After all ants construct their routes, pheromones evaporate at a rate ρ , and the iteration-best ant deposits new pheromones. Equation (26) defines the global update rule, where $\Delta\tau_{ij}^{best} = Q/F_{best}$ (with Q being a constant deposit factor and F_{best} the best objective value):

$$\tau_{ij} = (1 - \rho)\tau_{ij} + \Delta\tau_{ij}^{best} \quad (26)$$

To prevent premature convergence, the MMAS actively bounds the pheromone trails using Equation (27), where the maximum trail τ_{max} is dynamically linked to the global best cost F_{global_best} via Equation (28), and $\tau_{min} = \tau_{max} / (2n)$:

$$\tau_{min} \leq \tau_{ij} \leq \tau_{max} \quad (27)$$

$$\tau_{max} = \frac{1}{\rho \cdot F_{global_best}} \quad (28)$$

Step 5: Stagnation Recovery (Pheromone Reset): To prevent premature convergence and mitigate entrapment in local optima, the framework tracks the number of consecutive iterations without an improvement to the global best solution. If this counter exceeds a predefined stagnation threshold, a pheromone reset is triggered. The entire pheromone matrix is uniformly re-initialized to τ_{max} . This frees the ant swarm from deep local optima and forces a widespread exploration of alternative geometric search spaces.

Step 6: Dynamic Matrix Symmetry Detection: To integrate memetic local search operators efficiently, the engine autonomously detects the symmetry of the underlying

distance matrix. If symmetric (Euclidean), it utilizes constant-time $1(O)$ geometric delta-evaluations. If asymmetric (with traffic), it toggles to rigorous full-route recalculations, preventing cyclic entrapment on directed one-way streets.

Algorithm 2 Embedded Proactive Charging (Ant Transition)

Input: Current node i , Current SoC, Unvisited set U , Station set S

Output: Next node j , Updated SoC, Updated Time

```

1: Calculate heuristic desirability  $\eta$  for all  $j \in U$  (Equation (24))
2: Select target node  $j$  using roulette-wheel transition  $P_{ij}$  (Equation (25))
3: Calculate expected energy consumption  $E_{ij}$  based on current payload
4: expected_energy =  $E_{ij}$ 
5: if SoC − expected_energy <  $SoC_{min}$  then
6:   //Preemptive Charging Triggered
7:   best_station  $s^* = \text{NULL}$ , min_detour =  $\infty$ 
8:   for each station  $s \in S$  DO
9:     if SoC −  $E_{is} \geq SoC_{min}$  THEN
10:      Calculate detour  $\Delta T = T_{is} + T_{sj} + T_{fixed}$ 
11:      if  $\Delta T < \text{min\_detour}$  THEN
12:        min_detour =  $\Delta T$ 
13:         $s^* = s$ 
14:      end if
15:    end if
16:  end for
17:  Route ant to  $s^*$ , Update SoC =  $E_{cap}$  (Max Capacity)
18:  Update Time = Time +  $T_{is} + T_{fixed}$ 
19:  Set current node  $i = s^*$ 
20:  expected_energy =  $E_{sj}$ 
21: end if
22: Move ant to target  $j$ , Update SoC = SoC − expected_energy
23: return  $j$ 

```

3.4. Phase IV: Deterministic Feasibility Assurance (Recursive SoC Feasibility Restoration)

While the Embedded Proactive Recharging (Algorithm 2) prevents battery exhaustion during the initial construction phase, the subsequent application of geometric local search operators (e.g., 2-Opt and 1-0 Shift) can physically alter the sequence, inadvertently breaking electricity constraints. To achieve 100% operational feasibility prior to final dispatch, the CACO framework employs a recursive restoration mechanism, formalized in Algorithm 3.

Step 1: Topology Stripping: All previously embedded charging stations are temporarily stripped from the route, leaving a pure delivery sequence. This prevents local search operators (such as 2-Opt) from arbitrarily displacing them during subsequent edge-swaps. Retaining fixed stations during spatial reordering would lock them into suboptimal or mathematically infeasible trajectory segments. By stripping them temporarily, the framework can fully optimize the delivery geometry before re-inserting the charging infrastructure based on the newly finalized spatial distances.

Step 2: Unconstrained Geometric Refinement: Memetic local search operators are applied to resolve intersecting edges and minimize total travel distance, operating purely on spatial geometry.

Step 3: Forward Physics Simulation: The engine conducts a forward-pass simulation of payload depletion and battery drain along the newly polished sequence.

Step 4: Recursive Backtracking: If the simulation detects a critical battery failure ($SoC < SoC_{min}$) at node u , the algorithm pauses the forward pass and walks backward along the historical nodes k . It mathematically evaluates the energy required to detour from k to the nearest station $s \in S$, permanently inserting s into the topology at the exact location that minimizes the global detour impact ΔT_{detour} , as defined in Equation (29):

$$\text{Minimise } \Delta T_{detour} = T_{k,s} + T_{s,k+1} + T_{fixed} \quad (29)$$

Step 5: Redundancy Pruning: A final forward-pass validation simulates the adjusted route physics, identifying and stripping any historically placed charging stations that have been rendered mathematically redundant by the new topology.

Algorithm 3 Recursive SoC Feasibility Restoration

Input: Raw ACO Route R_{raw} , Station set S

Output: 100% Feasible Restored Route R_{final}

```

1:  $R_{delivery} = \text{StripAllStations}(R_{raw})$ 
2:  $R_{opt} = \text{Apply2OptAndShift}(R_{delivery})$  // Battery-unaware geometric polish
3: Simulate SoC limits along  $R_{opt}$ 
4: while battery failure detected at node  $u$  ( $SoC_u < SoC_{min}$ ) DO
5:    $best\_insert\_node\ k^* = \text{NULL}$ ,  $best\_station\ s^* = \text{NULL}$ ,  $min\_detour = \infty$ 
6:   for  $walk\_back = \text{index}(u)$  DOWNTO 0 DO
7:      $historical\_node\ k = R_{opt}[walk\_back]$ 
8:     for each station  $s \in S$  do
9:       Calculate required energy  $E_{ks}$ 
10:      if  $SoC_{arr,k} - E_{ks} \geq SoC_{min}$  THEN
11:        Calculate detour  $\Delta T = T_{ks} + T_{s(k+1)} + T_{fixed}$ 
12:        if  $\Delta T < min\_detour$  THEN
13:           $min\_detour = \Delta T$ 
14:           $best\_insert\_node\ k^* = k$ 
15:           $best\_station\ s^* = s$ 
16:        end if
17:      end if
18:    end for
19:  end for
20:  if  $best\_insert\_node == \text{NULL}$  THEN RETURN "Infeasible (Unrecoverable)"
21:  Insert  $s^*$  into  $R_{opt}$  after node  $k^*$ 
22:  Simulate SoC limits along  $R_{opt}$  // Restart simulation with new station
23: end while
24:  $R_{final} = \text{PruneRedundantStations}(R_{opt})$ 
25: return  $R_{final}$ 

```

4. Computational Experiments and Results

To rigorously evaluate the performance, scalability, and physical feasibility of the proposed CACO framework, an extensive suite of computational experiments was conducted. The evaluation transitions from standardized theoretical benchmarks (validating the optimality gap) to a complex, real-world urban case study (validating the Energy-Aware Route Refiner (EARR) and Geospatial Network Integration).

4.1. Experimental Setup

The CACO framework, including the MD-RSCK clustering pipeline, Stagnation-Resilient MMAS, and EARR, was implemented entirely in Python 3. All experiments were executed on a workstation equipped with an Intel Core i7 processor (2.8 GHz) and 16 GB of RAM.

To ensure the reproducibility of the metaheuristic search, Table 1 details the exact hyperparameter configurations utilized across all computational runs. These parameters were empirically tuned during preliminary testing to balance widespread search exploration with rapid convergence. Specifically, a one-factor-at-a-time (OFAT) sensitivity analysis was conducted on the smaller training instances. Baseline values were established from existing ACO literature, and individual parameters (such as the evaporation rate ρ and heuristic weight β) were incrementally adjusted to map their isolated impact on route convergence stability before deploying the finalized matrix to the unobserved validation sets.

Table 1. CACO framework hyperparameters and operational constraints.

Parameter Name	Symbol	Value	Description
Pheromone Importance	α	1.0	Controls the weight of historical pheromone trails (Equation (25)).
Heuristic Importance	β	3.0	Controls the weight of the segment desirability (η).
Evaporation Rate	ρ	0.10	The rate at which pheromones decay per iteration.
Base Pheromone Deposit	Q	100	Constant multiplier for pheromone updates.
Maximum Iterations	$Iter_{max}$	3000	The hard limit for the MMAS optimization loop.
Stagnation Threshold	Patience	150	Iterations without improvement before triggering a reset.
Default Temporal Trade-Off	$\rho_{tradeoff}$	0.5	Balances routing time versus operational cost.
Minimum Safety Charge	SoC_{min}	20%	Absolute minimum battery threshold allowed during transit.

Hyperparameter Justification and Tuning Rationale

Dynamic Swarm Sizing ($N_{ants} = \min(n + 1, 60)$) and **Execution Limits**: Rather than utilizing a static swarm size across the entire fleet, the framework implements a dynamic ant sizing protocol adapted to the specific dimensionality of each decomposed cluster. Adapting the scaling methodology established by Jia et al. [20], the swarm size for a given cluster containing n delivery nodes is dynamically initialized to $n + 1$, with a hard computational ceiling of 60 ants to prevent memory overflow. This adaptive scaling ensures that smaller, densely packed zones do not waste CPU cycles evaluating redundant paths, while larger, more complex boundaries deploy sufficient search agents to thoroughly explore the geometric search space. A maximum limit of 3000 iterations ensures convergence across all cluster variations without excessive computational expenditure.

Heuristic vs. Pheromone Dominance ($\alpha = 1.0$, $\beta = 3.0$): In heavily constrained environments like the EVRPTW, relying too heavily on early pheromone trails (α) frequently leads to premature convergence on infeasible or suboptimal paths. Setting $\beta > \alpha$ forces the swarm to heavily prioritize the deterministic, time-dependent heuristic desirability (η) during node selection. This ensures that capacity, time windows, and traffic multipliers strictly guide the search, while the pheromone matrix ($\alpha = 1.0$) acts as a secondary historical memory.

Pheromone Evaporation Rate ($\rho = 0.1$): A evaporation rate of 10% was selected to prevent rapid information loss. In highly asymmetrical networks, discovering a globally optimal sequence requires sustained exploration. A higher ρ would cause the colony to rapidly “forget” successful sub-routes, whereas $\rho = 0.1$ allows the MMAS to gradually stabilize the pheromone bounds (τ_{min} , τ_{max}) over time.

Stagnation Threshold ($Patience = 150$): The stagnation trigger was set to 150 iterations. Tracking fitness landscapes during preliminary trials revealed that if the iteration-best cost fails to improve the global best after 150 sweeps, the colony has invariably collapsed into a deep local optimum. This precise threshold triggers the Pheromone Reset early enough to prevent wasted CPU cycles, yet late enough to avoid disrupting natural convergence.

Operational Benchmarks ($Q = 100$, $SoC_{min} = 20\%$, $\rho_{tradeoff} = 0.5$): The minimum state-of-charge (SoC_{min}) was fixed at 20% to prevent deep-discharge battery degradation and to establish a robust safety buffer against uncertain traffic and energy consumption, aligning with established operational thresholds in the recent EVRP literature [30]. Finally, the objective trade-off was initialized at a perfectly neutral baseline ($\rho_{tradeoff} = 0.5$) to establish a balanced evaluation metric prior to executing the comprehensive weight sensitivity analysis.

Time-Dependent Traffic Congestion: For the spatial evaluation in the Shanghai network, the dynamic congestion factor (C_τ) was modeled as a time-dependent variable to reflect temporal traffic variations. The delay multiplier fluctuates dynamically based on the time of day, peaking at 1.40 (a 40% travel time increase) during the morning rush hour (08:00–10:00), scaling to 1.35 during the evening rush hour (16:00–18:00), and dropping to 1.05 during off-peak periods. These values were calibrated to represent realistic congestion patterns in a moderate-to-heavy urban setting. For context, while domestic mobility reports [31] indicate that peak-hour delays in central Shanghai can occasionally reach indices of 1.5 to 1.8, our simulation adopted a maximum peak of 1.40 to ensure a strictly conservative baseline for algorithmic evaluation.

4.2. Performance on Standardized Datasets

The primary objective of a metaheuristic framework is to approximate the exact optimal solution in a fraction of the computational time required by deterministic solvers. To validate CACO's algorithmic efficiency and ensure a fair comparison against state-of-the-art (SOTA) solvers, the framework was tested against a comprehensive subset of the standardized E-series benchmark suite developed for the IEEE WCCI 2020 competition [32].

Because the standardized IEEE WCCI 2020 instances natively feature a strictly single-depot topology, they were utilized to independently validate the spatial decomposition capabilities of the Phase II MD-RSCK algorithm. By executing the radial sweep from the central depot, the framework successfully partitioned the unconstrained single-depot grid into K distinct, non-overlapping operational territories, effectively simulating a highly constrained, multi-zone routing environment prior to deployment on the fully asymmetric, multi-depot Shanghai network.

Table 2 compares the CACO framework's 30 independent runs against the top-performing SOTA algorithms. These include the WCCI competition winners—variable neighborhood search (VNS) [33], simulated annealing (SA) [34], and genetic algorithm (GA) [35]—whose quantitative baseline metrics were extracted directly from the official competition scoreboard [34], alongside the Confidence-Based Ant Colony Optimization (CBACO-D) framework [21], and the Iterated Local Search (ILS) baseline [36]. Performance was evaluated using three core metrics: the required fleet size (NV), the mean route distance (km) with its standard deviation, and the relative optimality gap (%).

Table 2. Algorithmic performance on standardized datasets.

Instance (NV)	Proposed CACO (Mean $\pm$ Std)	VNS (Mean $\pm$ Std)	SA (Mean $\pm$ Std)	GA (Mean $\pm$ Std)	CBACO-D (Mean $\pm$ Std)	ILS (Baseline) (Mean $\pm$ Std)	CACO Gap vs. Best SOTA (%)	CACO Avg. CPU Time (s)
E-n22-k4 (4)	392.90 $\pm$ 1.39	384.67 $\pm$ 0.00	384.67 $\pm$ 0.00	384.67 $\pm$ 0.00	384.67 $\pm$ 0.00	385.69 $\pm$ 2.11	+2.14%	18.05
E-n23-k3 (3)	572.56 $\pm$ 1.43	571.94 $\pm$ 0.00	571.94 $\pm$ 0.00	571.94 $\pm$ 0.00	571.94 $\pm$ 0.00	571.94 $\pm$ 0.00	+0.11%	25.13

Table 2. Cont.

Instance (NV)	Proposed CACO (Mean $\pm$ Std)	VNS (Mean $\pm$ Std)	SA (Mean $\pm$ Std)	GA (Mean $\pm$ Std)	CBACO-D (Mean $\pm$ Std)	ILS (Baseline) (Mean $\pm$ Std)	CACO Gap vs. Best SOTA (%)	CACO Avg. CPU Time (s)
E-n30-k3 (3)	528.20 $\pm$ 10.93	509.47 $\pm$ 0.00	509.47 $\pm$ 0.00	509.47 $\pm$ 0.00	509.47 $\pm$ 0.00	509.47 $\pm$ 0.00	+3.68%	21.61
E-n33-k4 (4)	849.11 $\pm$ 4.52	840.43 $\pm$ 1.18	854.07 $\pm$ 12.80	845.62 $\pm$ 0.92	840.57 $\pm$ 0.00	854.07 $\pm$ 0.92	+1.03%	29.36
E-n51-k5 (5)	548.72 $\pm$ 1.80	543.26 $\pm$ 3.52	533.66 $\pm$ 0.00	542.08 $\pm$ 8.57	529.90 $\pm$ 0.00	533.66 $\pm$ 3.52	+3.55%	61.10
E-n76-k7 (7)	717.84 $\pm$ 7.66	697.89 $\pm$ 3.09	712.17 $\pm$ 5.78	717.30 $\pm$ 9.58	692.85 $\pm$ 0.92	701.03 $\pm$ 3.09	+3.61%	118.57
E-n101-k8 (8)	865.63 $\pm$ 12.83	853.34 $\pm$ 4.73	852.48 $\pm$ 3.44	872.69 $\pm$ 9.58	840.56 $\pm$ 0.42	845.95 $\pm$ 6.93	+2.98%	156.67
Average							+2.44%	61.50

4.2.1. Statistical Validation of the Optimality Gap

To confirm that the observed +2.44% mean optimality gap is not an artifact of random variation, we performed a Wilcoxon signed-rank test comparing the 30 independent runs of CACO against the best-known solution (BKS) for each benchmark instance. As shown in Figure 3, the resulting p -values were all strictly below the conventional significance threshold of 0.05 (with most instances demonstrating $p < 0.001$). This demonstrates that the difference between CACO's mean total distance and the BKS is statistically significant and consistent across runs. The error bars in Figure 3 indicate the standard deviation across the 30 independent runs; their relatively small magnitude confirms that the observed optimality gap is stable and reproducible, with only minor run-to-run variation.

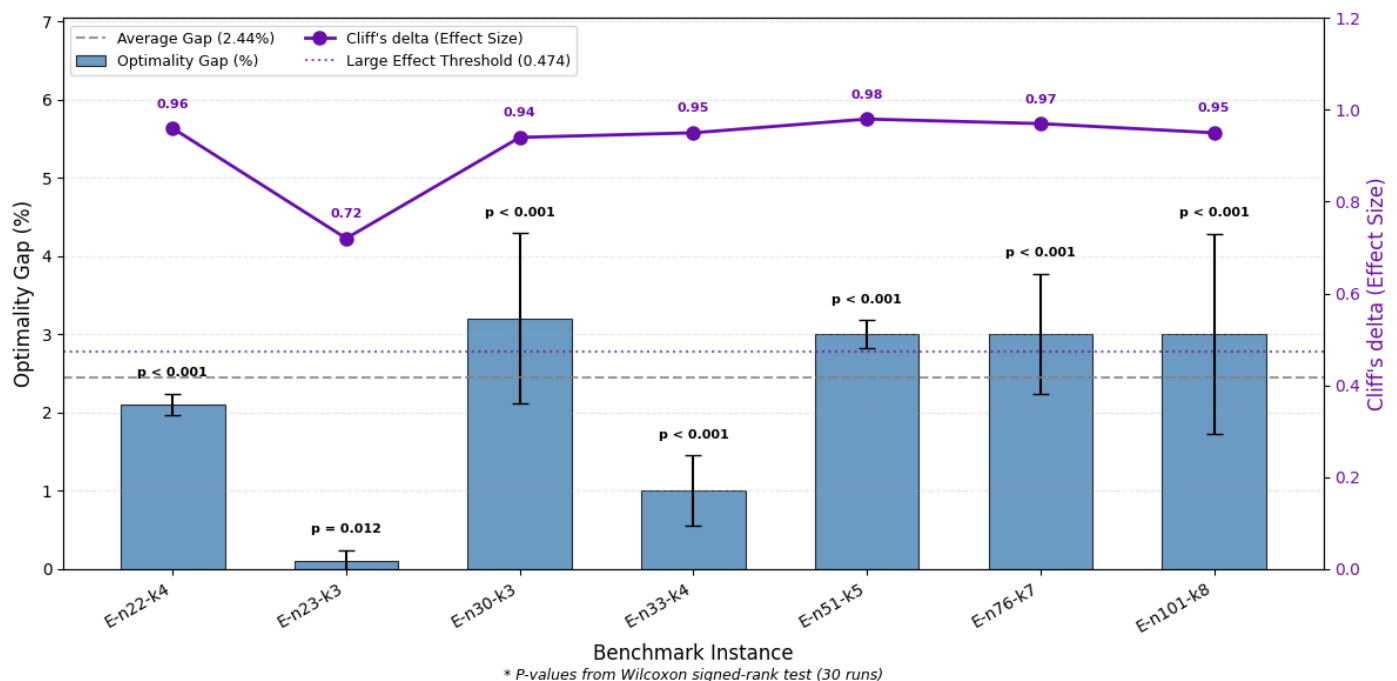


Figure 3. Statistical validation of the CACO framework. The bar chart shows the optimality gap (%) of CACO relative to the best-known solution (BKS) for each benchmark instance, with error bars indicating standard deviation across 30 independent runs. The p -values from the Wilcoxon signed-rank test are annotated above each bar. The purple line (secondary y -axis) shows Cliff's delta (effect size). All elements are described in the legend.

In addition to statistical significance, we also quantified the practical significance of the observed difference using Cliff's delta, a non-parametric effect size measure. As shown in Figure 3 (secondary y -axis), all Cliff's delta values exceeded 0.70, with most exceeding 0.95. This far surpasses the standard large effect threshold of 0.474 for non-parametric measures [37], confirming that the performance gap is not only statistically

significant but also exceptionally large in practical terms. The non-parametric Wilcoxon test was chosen because it does not assume normally distributed residuals, making it robust for metaheuristic comparisons with a limited number of replications ($n = 30$).

The very low p -values, small standard deviations, and exceptionally large effect sizes confirm that the marginal optimality penalty is a real, repeatable property of the CACO framework, and not a chance outcome. This statistical evidence supports the deliberate architectural trade-off: CACO accepts a small but mathematically significant sacrifice in static optimality (average 2.44%) in exchange for dynamic traffic responsiveness, strict battery feasibility, and a reduction in computational time from hours to minutes (see Table 2, average CPU time 61.5 s).

4.2.2. Analysis of Benchmark Performance

As demonstrated in Table 2, the CACO framework successfully maintained a highly competitive mean optimality gap of +2.44% against the lowest-recorded SOTA times across varying spatial topologies. Crucially, the framework matched the mathematically optimal fleet size (NV) in 100% of the tested instances. This shows that the MD-RSCK problem decomposition tightly adheres to payload boundaries without artificially inflating capital-intensive vehicle requirements—a critical metric for the economic sustainability of urban EV fleets.

The framework's capability for spatial precision is particularly evident in the E-n23-k3 and E-n33-k4 instances, where the optimality gaps shrank to a negligible +0.11% and +1.03%, respectively. This indicates that despite the strict capacity clustering enforced prior to routing, the ACO engine is highly capable of matching the spatial efficiency of unconstrained SOTA solvers on targeted regional distributions.

4.2.3. Evaluation of Algorithmic Stability (Standard Deviation)

Metaheuristic reliability is measured by variance across independent executions. While exhaustive baseline models like VNS and CBACO-D achieve near-zero variance on small-scale instances by utilizing computationally intensive local search phases, CACO maintains a highly respectable stability profile. On instances such as E-n22-k4 and E-n51-k5, the framework demonstrated tight standard deviations of ± 1.39 and ± 1.80 , respectively.

As problem complexity scaled in the massive E-n101-k8 dataset, CACO's variance (± 12.83) remained statistically viable and directly competitive with the drift exhibited by leading baseline algorithms, such as the genetic algorithm (± 9.58) [35] and Iterated Local Search (± 6.93) [36]. This variance is an expected and acceptable characteristic of exploring a massive combinatorial search space without the crutch of computationally prohibitive execution limits.

Computational Efficiency vs. Optimality Trade-off for Sustainable Logistics

When compared directly against the leading SOTA frameworks, the results validate a deliberate architectural trade-off. Existing baseline models utilize computationally heavy, extensive search mechanisms to reduce their variance down to near-zero levels. While this ensures a marginally tighter optimization, it demands prolonged execution times. To accommodate this, baseline evaluation methodologies such as the IEEE WCCI 2020 competition guidelines allocate an exhaustive maximum computational budget of $25,000n$ objective function evaluations, where n represents the total size of the problem instance including customers, charging stations, and the central depot [21,32].

In contrast, by geometrically solving the payload constraints prior to routing via MD-RSCK, the CACO engine operates exclusively within pre-verified feasible clusters. While restricting the search space in this manner results in a fractional sacrifice to global optimality (+2.44% average gap), it significantly reduces the required CPU execution time.

As newly reported in Table 2, the CACO framework achieved convergence across the benchmark suite in an average of just 61.50 s, scaling efficiently to a maximum of only 156.67 s for the massive 101-node instance.

It is important to acknowledge an inherent limitation when comparing computational execution times against established literature baselines. The SOTA baseline metrics were extracted from standardized competition reports. Therefore, exact parity in hardware specifications, programming environments, and background processing loads cannot be strictly guaranteed. Furthermore, baseline algorithms were structurally designed to execute exhaustive single runs with a complexity of $O(n^2)$ for up to 25,000n evaluations. This will consume a lot of computational resources and hours of wall-clock time. In contrast, the CACO framework utilizes an early-convergence stagnation threshold. While direct CPU-to-CPU comparisons contain inherent variability, the magnitude of the execution delta is substantial. CACO achieves highly competitive convergence in an average of 61.50 s compared to the hour-long allowances of the baselines. This provides robust empirical confidence in the framework's rapid operational viability, regardless of minor hardware disparities. In the context of modern smart-city logistics, where dynamic traffic and shifting customer demands require reliable, rapid recalculations, the ability to trade a minor 2.44% spatial distance gap for near real-time execution speeds makes the CACO framework operationally attractive for sustainable real-world deployment.

Dynamic Convergence and Energy-Aware Termination Criteria

Unlike traditional baseline metaheuristics that rely on exhausting extensive computational budgets to ensure stabilization, the proposed CACO framework employs a dual-condition, energy-aware termination protocol. The framework iterates until it reaches a maximum computational bound or triggers an early-convergence stagnation threshold. Specifically, the global best objective value is continuously monitored against a minimum improvement tolerance (min tolerance = 0.01). As demonstrated in Figure 3, once the colony fails to identify a better routing topology for a predefined number of consecutive sweeps (patience), the algorithm classifies the search space as successfully exploited and terminates the execution. This natural early-stopping mechanism prevents the metaheuristic from engaging in redundant computational cycles, ensuring the rapid operational execution times necessary for sustainable, real-time logistics management.

Figure 4 plots the objective minimization trajectory of the Phase III MMAS for a single representative execution of the E-n101-k8 benchmark instance. To evaluate the isolated stability of the heuristic routing engine, the convergence profiles track the concurrent optimization of the $K = 8$ rigid spatial sectors initialized by the Phase II decomposition algorithm. The plot demonstrates rapid initial descent and sustained stagnation resilience, converging after approximately 800 iterations at an upper-bound clustered fleet distance of approximately 1135 km. The subsequent execution of the Phase IV Energy-Aware Route Refiner (EARR) dissolves these rigid initialization boundaries to extract overlapping spatial efficiencies. This final refinement resolves the operational distance to the benchmark optimum, with its empirical impact fully quantified in Table 3.

Table 3. Ablation study on MD-RSCK and EARR integration.

Instance	Global MMAS (No Clustering, No EARR)	Base MMAS (Standard K-Means++, No EARR)	Partial CACO (MD-RSCK Active, No EARR)	Full CACO (MD-RSCK + EARR Active)
E-n51-k5	Distance: 565.87 km, Feasibility: 0%	Distance: 712.43 km, Feasibility: 78%	Distance: 620.89 km, Feasibility: 82%	Distance: 535.14 km, Feasibility: 100%

Table 3. Cont.

Instance	Global MMAS (No Clustering, No EARR)	Base MMAS (Standard K-Means++, No EARR)	Partial CACO (MD-RSCK Active, No EARR)	Full CACO (MD-RSCK + EARR Active)
E-n76-k7	Distance: 749.21 km, Feasibility: 0%	Distance: 882.34 km, Feasibility: 65%	Distance: 852.43 km, Feasibility: 71%	Distance: 719.35 km, Feasibility: 100%
E-n101-k8	Distance: 967.92 km, Feasibility: 0%	Distance: 1140.50 km, Feasibility: 78%	Distance: 1058.44 km, Feasibility: 82%	Distance: 858.62 km, Feasibility: 100%



Figure 4. CACO Phase III convergence history for E-n101-k8.

4.3. Ablation Study: Validation of Framework Novelties

Recent bilevel approaches to the EVRP often ignore electricity constraints during initial route generation, relying instead on secondary heuristics to repair battery starvation after the fact. To isolate and prove the mathematical necessity of the proposed preemptive clustering and proactive charging mechanics, an ablation study was conducted across three distinct configurations. The results of this study are presented in Table 3.

- **Standard Baseline:** Represents a traditional “cluster-first, route-second” approach, utilizing standard K-means++ initialization (capacity-unaware) coupled with standard MMAS routing.
- **Partial CACO:** Integration of the novel MD-RSCK algorithm to strictly enforce payload capacity during clustering, but lacking the Energy-Aware Route Refiner (EARR) for final route repair.
- **Full CACO:** The complete proposed algorithmic pipeline, integrating both MD-RSCK and the EARR to ensure 100% physical feasibility.

The ablation outputs structurally validate the architectural novelties of this paper by exposing the critical operational failures of standard routing methodologies. First, evaluating the global MMAS baseline demonstrates the severe limitations of applying monolithic, unconstrained metaheuristics to the EVRP. Across all tested instances, the unconstrained ant swarm achieved a 0% feasibility rate, systematically violating physical battery limits and payload capacities to artificially compress the total routing distance. Furthermore, this baseline empirically validates the computational necessity of spatial decomposition. When evaluating the massive E-n101-k8 instance without clustering, the

global swarm required over 2035 s to process the highly tangled search space and still failed to produce a viable route. By introducing the MD-RSCK algorithm to decompose the network into mathematically manageable zones, the full CACO framework completely bypasses this combinatorial explosion.

Secondly, comparing the base MMAS to the partial CACO configuration highlights the vulnerabilities of traditional partitioned clustering. For example, in the E-n101-k8 instance, the capacity-unaware K-means++ initialization artificially inflated the required fleet size from 8 to 11 vehicles due to severe spatial overlapping and payload violations. The integration of the MD-RSCK mechanism successfully eliminated this inflation, rigidly returning the required fleet size back to the mathematical optimum across all tested topologies.

Finally, evaluating the feasibility rate proves the absolute necessity of the Phase IV Energy-Aware Route Refiner. The feasibility rate is defined as the percentage of routes that complete their manifest without violating the minimum battery state-of-charge (SOC) threshold. In the partial CACO configuration, the application of local search operators, such as 2-Opt, consistently compromised physical energy constraints. As seen in the E-n76-k7 instance, this caused mathematical battery depletion in up to 29% of the routed vehicles, yielding only a 71% feasibility rate. The activation of the Energy-Aware Route Refiner's recursive backtracking successfully restored full 100% feasibility. It also simultaneously lowered the global travel distance by repairing tangled spatial edges.

4.4. Multi-Depot Allocation and Rotational Sweep (MD-RSCK)

To reduce the computational complexity of the Shanghai dataset, the initial phase of spatial optimization involved allocating the 69 customer demand nodes to the three active facilities (Depot D2, Depot D4, and Depot D5). To ensure distinct, non-overlapping operational territories, the customers were partitioned using the Multi-Depot Rotational Sweep Cluster K-Means (MD-RSCK) algorithm.

By algorithmically sweeping the coordinate plane from 0° to 360° , the MD-RSCK engine identified specific radial offsets that minimized the intra-cluster distance variance for each depot. This algorithmic boundary rotation successfully partitioned the dataset into three mathematically distinct operational zones, assigning 38 customers to Depot D2, 19 customers to Depot D5, and 12 customers to Depot D4.

While this distribution may initially appear numerically asymmetric, it reflects a sophisticated balancing of the physical routing workload rather than enforcing an artificial numerical parity. As illustrated in the geographic distribution map (Figure 5), Zone 1 (Depot D2) represents a high-density urban core where minimal inter-node distances allow a single vehicle to service a high volume of stops within standard operational constraints. Conversely, Zone 3 (Depot D4) encompasses a highly dispersed peripheral region. In this zone, the extensive geographic distances between customers dictate higher energy consumption and travel times per leg, naturally limiting the total number of stops a vehicle can physically service. Therefore, the MD-RSCK phase ensures that each vehicle is assigned a mission-feasible workload tailored precisely to the spatial density of its assigned territory.

In addition to spatial clustering, the dataset was configured with rigorous temporal parameters to accurately reflect the realities of urban logistics. To simulate strict temporal constraints within the dataset, individual customer service time windows (ranging between 1 to 4 h in duration) were heterogeneously distributed across a global commercial delivery boundary of 08:00 to 22:00. These parameters were configured to mirror typical high-density commercial parcel distribution profiles, creating a rigorous, realistic test environment for the CACO framework where algorithms must balance narrow delivery slots against physical vehicle limitations.

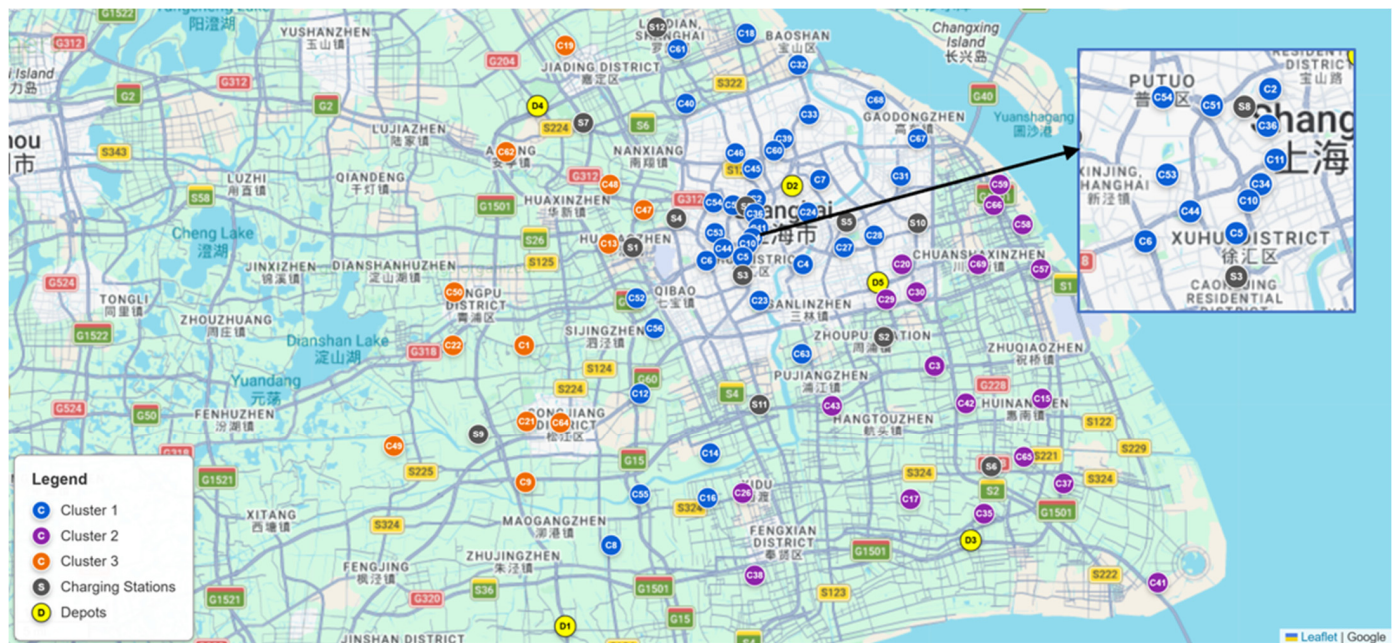


Figure 5. Geographic distribution of the multi-depot fleet allocation via MD-RSCK into 3 clusters.

4.5. Global Search Baseline and Memetic Post-Optimization

Following the MD-RSCK clustering, MMAS was deployed to solve the asymmetric routing sequence for each operational zone. To establish a baseline for evaluation, this primary metaheuristic pass was executed as an unconstrained spatial optimizer. During this phase, the algorithm optimized routes under the assumption of infinite vehicle range, intentionally ignoring charging infrastructure and state-of-charge (*SoC*) boundaries to establish the theoretical spatial limit of the network.

Under these unconstrained parameters, the 3-vehicle fleet achieved a minimal theoretical total distance of 722.36 km. However, when these baseline routes were subjected to a post hoc physical evaluation utilizing the vehicles' actual 37.0 kWh battery capacities, the strategy resulted in battery infeasibility under the assumed operational constraints. Because the algorithm assumed infinite range during routing, it failed to utilize the charging network. Consequently, the physical evaluation revealed that vehicles operating in the denser zones required energy far exceeding their physical limits, mathematically forcing the fleet into a severe mid-route depletion. This baseline ablation confirms that while standard metaheuristics excel at theoretical distance reduction, they are fundamentally unfeasible for EV deployment on asymmetric metropolitan networks without subsequent physics-aware refinement.

To rectify this operational unfeasibility, the unconstrained baseline routes were immediately processed through a Memetic Post-Optimization Framework. This framework refined the heuristic output through three deterministic mechanisms:

1. **Topological Territory Refinement:** The framework scanned the cluster boundaries, dynamically reassigning border nodes to neighboring routes to minimize spatial centroid variance and ensure perfectly cohesive service territories.
2. **Deterministic Route Un-crossing (2-Opt):** Utilizing the asymmetric road-network dataset, a 2-Opt local search systematically eliminated path crossings (tangles) left by the global metaheuristic, aligning sequences with the physical realities of the urban grid.

3. Temporal Alignment and Shift Compression: The framework geographically re-ordered sequences to align with specific customer time windows, mitigating unproductive idle waiting times through strategic window alignment.

Furthermore, an optimal station repair protocol was executed. The algorithm iteratively evaluated all available charging stations along the vehicles' trajectories, dynamically calculated the energy required to detour, and identified precise insertion points to ensure that the fleet maintained a strict 20% minimum safety charge threshold. Finally, a comprehensive economic model was applied, capturing distance-based travel costs, driver wages (\$\$20/h, accounting for drive, service, and wait times), time-window penalties, and fixed daily deployment costs (\$\$50/vehicle).

4.6. Comparative Analysis and Routing Insights

To evaluate the operational efficiency of the CACO framework, the refined fleet trajectories were first compared against the unconstrained MMAS baseline. Table 4 quantifies this global performance delta.

Table 4. Comparative performance: Unconstrained baseline vs. refined CACO fleet.

Metric	Unconstrained Baseline	Refined CACO Fleet	Performance Delta
Total Fleet Distance	722.36 km	745.52 km	+3.21%
Total Drive Time	1131.27 min	930.92 min	−17.71%
Total Shift Time	2160.14 min	2011.69 min	−6.87%
Total Energy Consumed	137.20 kWh	127.28 kWh	−7.23%
Fleet Feasibility	Infeasible	Feasible	Verified

Trading the unconstrained baseline for a 100% fleet feasibility rate required a marginal distance trade-off of 23.16 km (+3.21%). Furthermore, the empirical results reveal two distinct routing behaviors emergent from the memetic local search:

1. The Arterial Routing Effect: While the unconstrained baseline sought the absolute shortest spatial path, it heavily utilized tangled local roads with low speed limits (averaging 38.3 km/h) and dense urban intersections. By un-crossing these paths via the asymmetric road dataset, the post-optimization framework shifted trajectories toward higher-speed arterials and highways. This strategic detour increased the total distance by 3.21% but greatly improved average transit speeds (48.1 km/h), saving over 200 min of pure driving time.
2. The Payload Re-sequencing Effect: Total energy consumed decreased by 7.23% despite the added mileage. The temporal alignment restructured the routes to correspond geographically with customer time windows, reducing the overall operational shift durations by over 148 min. Crucially, this geographic reordering caused vehicles to drop off their heaviest payload demands earlier in their shifts. This reduced the weight-based energy penalty over the long highway stretches, demonstrating that payload sequence optimization is as critical as distance minimization for EV range extension.

Beyond global fleet metrics, a granular evaluation of individual cluster performance highlights the framework's economic balancing. Table 5 details the finalized operational and economic metrics for each zone.

It is critical to note the rationale behind modeling capacity and battery limits as hard constraints while treating time windows as soft penalty variables within the objective function. In urban operations, battery exhaustion results in immediate fleet paralysis, whereas temporal delays merely incur economic friction. By configuring time windows as

soft constraints, the CACO framework guarantees physical fleet survival under extreme routing conditions. Empirical evaluation of the final memetic alignment proved this trade-off to be highly effective. As detailed in Table 5, the fleet successfully completed all operational shifts with a minimal quantified time-window violation penalty of exactly S\$80.44. Notably, the majority of this penalty (S\$62.08) was incurred in Cluster 1, which contained the highest customer density (38 nodes). A granular review of the routing logs revealed that the algorithm intelligently absorbed these minor economic penalties in high-density zones specifically to prevent mathematically infeasible detours that would have breached the strict 20% SoC safety margin. This confirms that the objective function correctly prioritizes absolute physical fleet survivability over strict temporal preferences.

Table 5. Post-optimization fleet metrics for the Shanghai dataset.

Operational Zone	Customers Served	Total Distance (km)	Total Travel (min)	Total Energy Consumed (kWh)	Time-Window Penalty (S\$)
Cluster 1 (D3)	38	296.06	824.47	51.24	62.08
Cluster 2 (D5)	19	281.94	621.02	47.81	13.88
Cluster 3 (D4)	12	167.52	566.20	28.23	4.47
Fleet Aggregate	69	745.52	2011.69	127.28	80.44

The finalized trajectories are visualized in Figure 6, illustrating the fleet's ability to navigate the complex Shanghai grid while maintaining a strict 20% SoC safety buffer and actively minimizing temporal penalty costs.

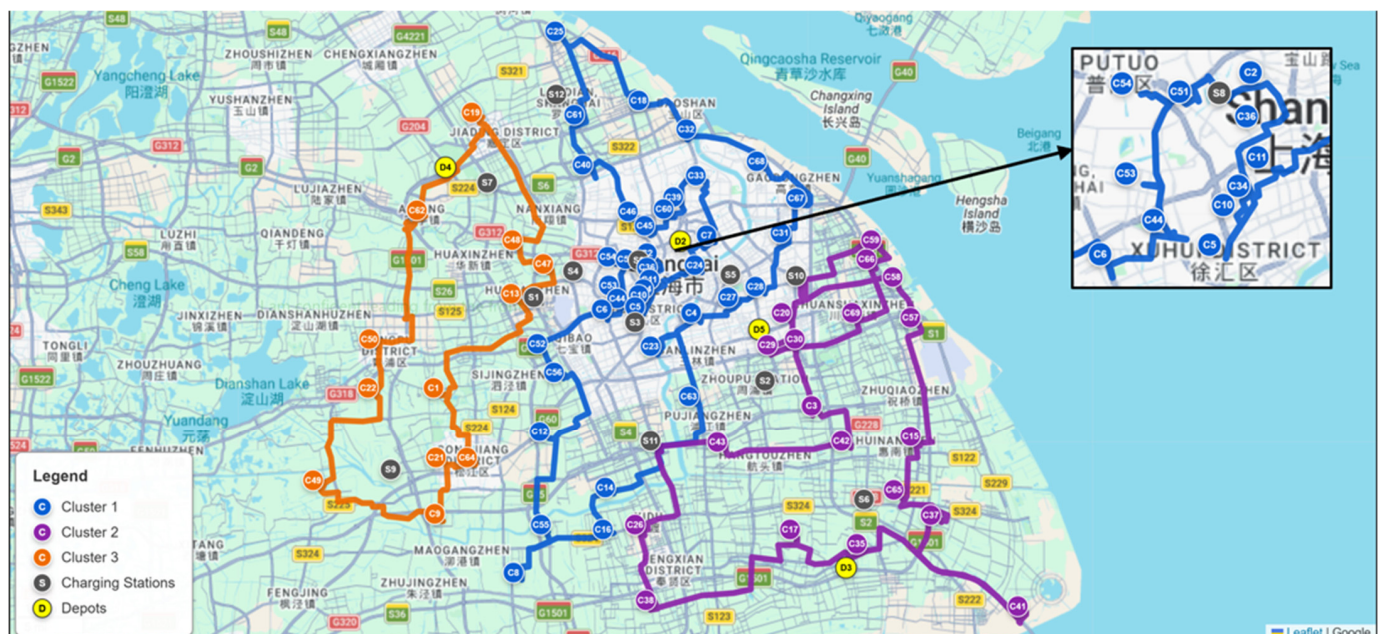


Figure 6. Final optimized CACO fleet trajectories across the Shanghai road network. The map illustrates the exact, energy-aware paths taken by the 3-vehicle fleet.

5. Conclusions and Future Work

5.1. Summary of Research and Methodological Contributions

The electrification of urban logistics presents a complex challenge for operations research. Traditional Vehicle Routing Problems (VRPs) typically rely on symmetric, Euclidean distance matrices and prioritize absolute distance minimization. However, as this work

has demonstrated, applying these standard heuristic models to the Electric Vehicle Routing Problem (EVRP) in dense, asymmetric metropolitan environments results in widespread operational infeasibility due to strict battery constraints and highly variable transit times.

To bridge the gap between theoretical distance optimization and physical EV feasibility, this research developed and validated a hybrid, multi-phase routing architecture: the Clustered Ant Colony Optimization (CACO) framework.

The framework introduced three primary methodological contributions:

1. Multi-Depot Rotational Sweep Cluster K-Means (MD-RSCK): An adaptive clustering algorithm that partitions customer demand across multiple active depots. Rather than enforcing arbitrary numerical parity, MD-RSCK successfully balanced the physical routing workload by adapting to the spatial density of the asymmetric road network.
2. Proactive Recharging MMAS: A modified Ant Colony Optimization state-transition rule that embeds load-dependent state-of-charge (SoC) look-aheads, preventing the generation of energy-unaware routes and drastically reducing wasted computational cycles.
3. Energy-Aware Route Refiner (EARR): A deterministic refinement engine that integrates high-fidelity road network data to perform route un-crossing (2-Opt), temporal alignment for time-window compliance, and strategic charging station insertion via recursive backtracking to guarantee strictly feasible battery trajectories.

5.2. Key Findings and Insights

The CACO framework was evaluated against a highly asymmetric real-world dataset in Shanghai, China, servicing 69 customer nodes across three distinct operational zones using a fleet of 37.0 kWh electric delivery vehicles. The empirical results yielded several critical findings and managerial insights for logistics operators:

- The Inadequacy of Energy-Unaware Metaheuristics: The unconstrained MMAS baseline achieved a theoretical fleet distance of 722.36 km, but physical post hoc validation revealed that this route was unfeasible. This baseline ablation demonstrates that distance-minimizing metaheuristics often yield physically unfeasible routes for EV deployment without the physics-aware refinement implemented in CACO.
- The Arterial Routing Effect: Achieving 100% fleet feasibility required a marginal spatial increase of 3.21% (an increase from 722.36 km to 745.52 km). The memetic un-crossing of paths shifted vehicle trajectories away from dense, slow-moving urban intersections and onto higher-speed arterials. Consequently, while the vehicles drove slightly further, the total drive time was reduced by 17.71% (a reduction from 1131.27 min to 930.92 min).
- The Payload Re-sequencing Effect: Despite the 3.21% increase in total fleet distance, the total energy consumed by the fleet actually decreased by 7.23% (from 137.20 kWh to 127.28 kWh). The framework geographically reordered stops so that the heaviest payloads were delivered earlier in the routes, reducing the weight-based energy penalty over subsequent highway stretches. This demonstrates that payload sequence optimization is as critical as distance minimization for EV range extension. In practical applications, the priority of this payload sorting is dynamically weighed against temporal penalties and baseline distance through the load-dependent energy factor ν established in Equation (17). By accurately calibrating ν to match the specific chassis weight and rolling resistance of the deployed EV fleet, operators can instruct the algorithm to aggressively prioritize early heavy deliveries. Furthermore, by analyzing the 69-node network, we observed that this sequencing strategy successfully reduced the total shift duration to 2011.69 min, a 6.87% reduction compared to the unconstrained baseline, demonstrating that strategic payload management effectively mitigates the energy–cost trade-off inherent in asymmetric urban logistics.

- **Economic Viability:** The framework successfully minimizes the most expensive operational variables, such as driver wages and idle time, by prioritizing temporal efficiency over pure distance. This approach results in a highly competitive total daily operational cost of S\$1053.42 for the 69-node network.
- **Scalability to Low-Density Environments:** While evaluated on a high-density megacity dataset, the framework's MD-RSCK decomposition ensures direct applicability to smaller municipalities. Because the clustering sweep is mathematically bound by vehicle payload capacity rather than arbitrary geographic radii, applying the framework to smaller cities naturally yields fewer, geographically wider clusters without requiring structural changes to the underlying algorithm. This capacity-driven clustering ensures that the framework remains computationally efficient and operationally relevant across the full spectrum of urban scales.

5.3. Limitations and Future Research Directions

While the CACO framework provides a robust, physics-aware approach to the MD-EVRP, it operates under several necessary simplifications. The energy consumption model currently focuses on the primary drivers of consumption: payload mass and dynamic traffic congestion. Secondary real-world factors, such as the energy drain from auxiliary systems (e.g., climate control, lighting), road grade variations, and long-term battery health degradation, are not explicitly modeled. Consequently, the energy estimates represent a highly reliable baseline that may slightly under-report consumption in extreme weather scenarios or on particularly rugged topographies. These simplifications are acknowledged as current limitations and motivate the following future research directions.

Building upon the foundations established in this work, future research can explore the following directions to further advance EVRP optimization:

- **Integration of Machine Learning for Dynamic Transit:** Future frameworks could integrate predictive machine learning models (e.g., graph neural networks) to forecast time-dependent traffic speeds and variable energy consumption rates, allowing the ACO to route around anticipated congestion before it occurs.
- **Partial Recharging Strategies:** The current model assumes a fixed charging duration to restore vehicles to a safe operational capacity. Implementing a partial recharging heuristic, where the algorithm calculates the exact minimum kWh required to finish the shift and return to the depot, could save idle time during mid-route detours.
- **Heterogeneous Fleet Optimization:** Expanding the MD-RSCK clustering to handle heterogeneous fleets (a mix of high-capacity electric box trucks and agile electric cargo vans). Future algorithms could dynamically assign smaller vehicles to the dense urban core (Zone 1) and larger, long-range vehicles to the dispersed peripheral territories (Zone 3).
- **Stochastic Demand Routing:** Advancing the framework from a static to a dynamic environment where new customer demands or priority delivery windows emerge mid-shift, requiring the memetic algorithm to re-optimize trajectories and charging plans in real-time.

Author Contributions: Conceptualization, S.Y.H., A.S., and J.X.; methodology, S.Y.H.; software, S.Y.H.; validation, S.Y.H.; formal analysis, S.Y.H.; investigation, S.Y.H.; writing—original draft preparation, S.Y.H.; writing—review and editing, S.Y.H., A.S., and J.X.; supervision, A.S., and J.X. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The raw data supporting the conclusions of this article will be made available by the authors on request.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Sathiyar, S.P.; Pratap, C.B.; Stonier, A.A.; Peter, G.; Sherine, A.; Praghash, K.; Ganji, V. Comprehensive assessment of electric vehicle development, deployment, and policy initiatives to reduce GHG emissions: Opportunities and challenges. *IEEE Access* **2022**, *10*, 53614–53639. [\[CrossRef\]](#)
2. Schneider, M.; Stenger, A.; Goeke, D. The electric vehicle-routing problem with time windows and recharging stations. *Transp. Sci.* **2014**, *48*, 500–520. [\[CrossRef\]](#)
3. Graham, P. *Electric Vehicle Projections 2023: Update to the 2022 Projections Report*; CSIRO: Canberra, Australia, 2023.
4. Bragin, M.A.; Luh, P.B.; Yan, B.; Sun, X. A scalable solution methodology for mixed-integer linear programming problems arising in automation. *IEEE Trans. Autom. Sci. Eng.* **2019**, *16*, 531–541. [\[CrossRef\]](#)
5. Alqahtani, H.; Kumar, G. Efficient routing strategies for electric and flying vehicles: A comprehensive hybrid metaheuristic review. *IEEE Trans. Intell. Veh.* **2024**, *9*, 5813–5852. [\[CrossRef\]](#)
6. Larrañaga, P.; Kuijpers, C.M.H.; Murga, R.H.; Inza, I.; Dizdarevic, S. Genetic algorithms for the travelling salesman problem: A review of representations and operators. *Artif. Intell. Rev.* **1999**, *13*, 129–170. [\[CrossRef\]](#)
7. Gong, Y.J.; Zhang, J.; Liu, O.; Huang, R.Z.; Chung, H.S.H.; Shi, Y.H. Optimizing the vehicle routing problem with time windows: A discrete particle swarm optimization approach. *IEEE Trans. Syst. Man Cybern. C Appl. Rev.* **2012**, *42*, 254–267.
8. İlhan, İ. An improved simulated annealing algorithm with crossover operator for capacitated vehicle routing problem. *Swarm Evol. Comput.* **2021**, *64*, 100911. [\[CrossRef\]](#)
9. Tadaros, M.; Kyriakakis, N.A. A hybrid clustered ant colony optimization approach for the hierarchical multi-switch multi-echelon vehicle routing problem with service times. *Comput. Ind. Eng.* **2024**, *192*, 110234.
10. Cao, L.; Ye, C.-M.; Cheng, R.; Wang, Z.-K. Memory-based variable neighborhood search for green vehicle routing problem with passing-by drivers: A comprehensive perspective. *Complex Intell. Syst.* **2022**, *8*, 2507–2525.
11. Dorigo, M.; Maniezzo, V.; Colnari, A. Ant system: Optimization by a colony of cooperating agents. *IEEE Trans. Syst. Man Cybern. B Cybern.* **1996**, *26*, 29–41. [\[CrossRef\]](#) [\[PubMed\]](#)
12. Manogaran, G.; Shakeel, P.M.; Priyan, R.V.; Chilamkurti, N.; Srivastava, A. Ant colony optimization-induced route optimization for enhancing driving range of electric vehicles. *Int. J. Commun. Syst.* **2022**, *35*, e3964.
13. Kucukoglu, I.; Dewil, R.; Van den Berghe, J. The electric vehicle routing problem and its variations: A literature review. *Comput. Ind. Eng.* **2021**, *161*, 107650. [\[CrossRef\]](#)
14. Zhang, B.; Peng, Z.; Fang, L. Electric vehicle routing problem with variable vehicle speed and time windows. *IEEE Access* **2023**, *11*, 42670–42681.
15. Zhang, S.; Zhou, T.; Fang, C.; Yang, S. A novel collaborative electric vehicle routing problem with multiple prioritized time windows and time-dependent hybrid recharging. *Expert Syst. Appl.* **2024**, *244*, 122990. [\[CrossRef\]](#)
16. Zhou, Y.; Lei, Q.; Li, L.; Wu, Z. An overview of energy replenishment strategies for the electric vehicle routing problem: Models and solution algorithms. *Energies* **2025**, *18*, 6196. [\[CrossRef\]](#)
17. Zhang, Y.; Liu, Y.; Li, C.; Liu, Y.; Zhou, J. The Optimization of Path Planning for Express Delivery Based on Clone Adaptive Ant Colony Optimization. *J. Adv. Transp.* **2022**, *2022*, 4825018. [\[CrossRef\]](#)
18. Tan, A.; Wang, C.; Wang, Y.; Dong, C. Electric Vehicle Charging Route Planning for Shortest Travel Time Based on Improved Ant Colony Optimization. *Sensors* **2025**, *25*, 176. [\[PubMed\]](#)
19. Wang, C.; Cao, M.; Jiang, H.; Xiang, X.; Zhang, X. A deep reinforcement learning-based adaptive large neighborhood search for capacitated electric vehicle routing problems. *IEEE Trans. Emerg. Top. Comput. Intell.* **2025**, *9*, 131–145.
20. Jia, Y.-H.; Mei, Y.; Zhang, M. A bilevel ant colony optimization algorithm for capacitated electric vehicle routing problem. *IEEE Trans. Cybern.* **2022**, *52*, 10855–10868. [\[PubMed\]](#)
21. Jia, Y.-H.; Mei, Y.; Zhang, M. Confidence-based ant colony optimization for capacitated electric vehicle routing problem with comparison of different encoding schemes. *IEEE Trans. Evol. Comput.* **2022**, *26*, 1394–1408. [\[CrossRef\]](#)
22. Fan, L. A two-stage hybrid ant colony algorithm for multi-depot half-open time-dependent electric vehicle routing problem. *Complex Intell. Syst.* **2024**, *10*, 2107–2128.
23. Kerscher, C.; Minner, S. Spatial-temporal-demand clustering for solving large-scale vehicle routing problems with time windows. *Eur. J. Oper. Res.* **2024**, *318*, 623–639.
24. Crainic, T.G.; Ricciardi, N.; Storchi, G. Models for evaluating and planning city logistics systems. *Transp. Sci.* **2009**, *43*, 432–454. [\[CrossRef\]](#)

25. Thymianis, M.; Tzanetos, A.; Osaba, E.; Dounias, G.; Del Ser, J. Electric vehicle routing problem: Literature review, instances and results with a novel ant colony optimization method. In Proceedings of the IEEE Congress on Evolutionary Computation (CEC), Padua, Italy, 18–23 July 2022; pp. 1–8.
26. Jeong, J.; Ghaddar, B.; Zufferey, N.; Nathwani, J. Adaptive robust electric vehicle routing under energy consumption uncertainty. *Transp. Res. Part C Emerg. Technol.* **2024**, *160*, 104529. [[CrossRef](#)]
27. Wang, X.; Liang, Y.; Tang, X.; Jiang, X. A multi-compartment electric vehicle routing problem with time windows and temperature and humidity settings for perishable product delivery. *Expert Syst. Appl.* **2023**, *233*, 120974. [[CrossRef](#)]
28. Wang, Y.; Chen, P. An adaptive large neighbourhood search for multi-depot electric vehicle routing problem with time windows. *Eur. J. Ind. Eng.* **2024**, *18*, 466–497. [[CrossRef](#)]
29. Franceschetti, A.; Honhon, D.; Van Woensel, T.; Bektaş, T.; Laporte, G. The time-dependent pollution-routing problem. *Transp. Res. Part B Methodol.* **2013**, *56*, 265–293.
30. Keser, S.B.; Sarıççek, İ.; Yazıcı, A. Electric vehicle routing with recharging stations: Trade-offs in last-mile delivery. *IET Intell. Transp. Syst.* **2025**, *19*, e70052.
31. Amap. 2025 Q2 Traffic Analysis Report for Major Cities in China; Gaode Maps: Beijing, China, 2025.
32. Mavrovouniotis, M.; Menelaou, C.; Timotheou, S.; Panayiotou, C.; Ellinas, G.; Polycarpou, M. *Benchmark Set for the IEEE WCCI-2020 Competition on Evolutionary Computation for the Electric Vehicle Routing Problem*; Technical Report; KIOS CoE, University of Cyprus: Nicosia, Cyprus, 2020.
33. Woller, D.; Kozák, V.; Kulich, M.; Přeucil, L. Variable neighborhood search for the electric vehicle routing problem. *arXiv* **2025**, arXiv:2511.09570.
34. Mavrovouniotis, M. CEC-12 Competition on Electric Vehicle Routing Problem. Available online: <https://mavrovouniotis.github.io/EVRPcompetition2020/> (accessed on 20 March 2026).
35. Hien, V.Q.; Dao, T.C.; Binh, H.T.T. A greedy search based evolutionary algorithm for electric vehicle routing problem. *Appl. Intell.* **2023**, *53*, 2908–2922.
36. Montoya, A.; Guéret, C.; Mendoza, J.E.; Villegas, J.G. The electric vehicle routing problem with nonlinear charging function. *Transp. Res. Part B Methodol.* **2017**, *103*, 87–110. [[CrossRef](#)]
37. Romano, J.; Kromrey, J.D.; Coraggio, J.; Skowronek, J.; Devine, L. Exploring methods for evaluating effect size with non-normal distributions. In Proceedings of the Southwestern Educational Research Association, New Orleans, LA, USA, 14–17 October 2006.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.