aerospace

MDPI

Article

A Multi-Objective Optimization Model for Collaborative UAV–Rider Meal Delivery Routing Using an Enhanced NSGA-II Algorithm

Shichen Hu [1], Yuan Zheng [2], Jingqi Dai [3,*], Hui Jiang [4] and Yiwu Feng [5]

[1] School of International Economics and Trade, Shanghai University of International Business and Economics, Shanghai 201620, China; hushichen@suibe.edu.cn
[2] School of Computer Science and Artificial Intelligence, Civil Aviation Flight University of China, Guanghan 618307, China; ranchozy@cafuc.edu.cn
[3] School of Economics and Management, Civil Aviation Flight University of China, Guanghan 618307, China
[4] School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China; jianghui@uestc.edu.cn
[5] Huadian Electric Power Research Institute Co., Ltd., Hangzhou 310030, China; feng_yiwu@outlook.com
* Correspondence: daijingqi@cafuc.edu.cn

Abstract

Urban instant-delivery platforms increasingly require efficient, punctual, and low-carbon delivery services. Unmanned aerial vehicles (UAVs) can reduce dependence on congested ground traffic in meal delivery and further improve overall delivery efficiency through coordinated operations with ground riders at rendezvous points. However, existing studies mainly focus on ground-based routing or simplify UAV-assisted delivery as a single-objective problem, limiting their ability to balance cost, completion time, carbon emissions, and service quality. To address this limitation, this paper investigates a collaborative UAV–rider meal delivery routing problem and formulates a multi-objective optimization model integrating restaurant pickup, UAV transfer, rider last-mile delivery, and soft customer time windows. An Enhanced NSGA-II algorithm is then developed, where hybrid initialization improves solution quality and diversity, adaptive operators balance exploration and exploitation, local search refines route structures, structural repair maintains feasibility, and diversity preservation supports a well-distributed Pareto front. Comparative experiments against NSGA-II, MOPSO, NSGA-III, MOEA/D, MODE, and IMODE, together with scalability, ablation, sensitivity, and case analyses, show that E-NSGA-II provides stronger Pareto-front approximation. The results support its use as a decision-support method for service-aware and low-carbon UAV–rider meal delivery, while also revealing additional computational cost.

Keywords: collaborative UAV–rider delivery; meal delivery; multi-objective optimization; enhanced NSGA-II; rendezvous assignment

Check for updates

Academic Editors: Volker Gollnick and Michael Schultz

Received: 3 July 2026
Revised: 24 July 2026
Accepted: 26 July 2026
Published: 28 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) license.

1. Introduction

An unmanned aerial vehicle (UAV), commonly known as a drone, is an unpiloted aircraft operated via radio control equipment or autonomous programs. It integrates multiple technologies such as communications, sensors, data processing, intelligent control, and aerospace propulsion, making it a high-tech product of the information age [1]. As an aerial platform, the core value of UAVs lies in their ability to integrate with other smart devices and extend human capacity to perform tasks in the air [2]. In recent years, with the

continuous advancement of UAV technology, their application in the logistics sector has become a new trend in the development of modern logistics, owing to advantages such as low cost, high speed, strong practicality, high flexibility, and the ability to operate unrestricted by terrain or roads [3].

With the rapid development of urban instant delivery and the catering industry, traditional ground-based delivery systems are facing challenges such as traffic congestion, high labor costs, and significant environmental impacts. In addition, stringent delivery-time requirements can create substantial time pressure for riders, which is associated with job strain, risk-taking attitudes, and lower compliance with road-safety regulations [4]. Meanwhile, the integration of the low-altitude economy with modern logistics systems is growing closer, moving toward intelligent, efficient, and green development [5]. Therefore, achieving fast and secure item delivery has become an urgent issue for the instant logistics industry in recent years. As a disruptive technology, drones have emerged to overcome these obstacles, enabling direct, point-to-point delivery, shortening transportation time, and reducing emissions.

However, the practical deployment of UAV delivery systems introduces routing and scheduling challenges that differ fundamentally from the conventional vehicle routing problem (VRP). In addition to travel distance, UAV routing must account for flight endurance time, payload capacity, no-fly zones, weather conditions, and energy consumption. To address these constraints, Calamoneri et al. formulated a multi-depot, multi-trip routing model for post-disaster logistics to minimize mission completion time [6], while Jung proposed a mixed-integer linear programming model to jointly optimize delivery cost and service time for cooperative UAV–UGV last-mile delivery [7]. Jasim and Fourati further developed a guided genetic algorithm that efficiently solves capacitated UAV routing problems under operational constraints [8]. Beyond routing, recent studies have also investigated multi-objective optimization for UAV task allocation and resource scheduling, balancing efficiency, communication quality, security, and operational cost [9,10].

From an algorithmic perspective, transportation and engineering optimization studies have progressed beyond standard NSGA-II toward parallel, decomposition-based, reference-direction, and differential-evolution approaches. Akopov et al. developed a parallel bi-objective real-coded genetic algorithm for a multi-agent fuzzy transportation system [11]. Zhang and Li introduced MOEA/D, which decomposes a multi-objective problem into coordinated scalar subproblems [12], while Deb and Jain proposed NSGA-III to preserve diversity through reference directions in many-objective search [13]. Xue et al. extended differential evolution through Pareto-based selection in MODE [14], and Sallam et al. developed the multi-operator adaptation strategy underlying IMODE [15]. More recently, He et al. incorporated orthogonal experimental design into a decomposition-based evolutionary algorithm to improve selection for irregular Pareto fronts [16]. These developments motivate comparisons with dominance-based, decomposition/reference-direction, and differential-evolution baselines in the present four-objective routing problem.

In the field of drone path planning, scholars primarily focus on two main research directions: single-drone mission execution and collaborative operations involving drones with humans, trucks, and other vehicles. On the one hand, scholars have proposed an integrated solution for remote visual inspection of overhead power lines using drones to achieve the shortest mission time. This solution effectively coordinates field operations—including route planning, autonomous execution, and monitoring—while meeting end-user requirements [17]. Besides, Divkoti and Nouri-Baygi addressed the problem of collision-free path planning for drones moving from a source to a target point within a polygonal domain [18]. On the other hand, Yang et al. introduced a bi-layer task-planning algorithm based on drone–human collaboration to enhance efficiency and positioning accuracy in search and

rescue missions. The algorithm consists of a search layer performed by drones and an execution layer carried out by rescue personnel [19]. Gonzalez-R et al. tackled the last-mile logistics delivery problem using a truck–drone team by proposing an iterative greedy heuristic based on a destruction-and-reconstruction solution process, with simulated annealing applied for global optimization [20]. Long et al. developed a dynamic truck–drone coordination strategy and designed a tabu search-based integrated scheduling algorithm to achieve efficient and resilient urban emergency response [21]. In summary, a core challenge in this research area lies in pursuing optimal solutions under multiple constraints, including limited flight distance, payload capacity, energy consumption, and complex environmental conditions.

Recent technological advances have primarily focused on addressing the two major limitations of UAV systems, namely limited flight endurance time and payload capacity. For instance, Kim and Lee proposed an intelligent battery management system combining active cell balancing with fuzzy logic control to improve battery utilization and extend operational lifetime [22]. Liao et al. developed an adaptive stochastic configuration network to accurately predict the remaining useful life of UAV batteries, enabling predictive maintenance and more reliable mission planning [23]. To overcome energy-density constraints, Joo et al. designed a hybrid propulsion system integrating an internal combustion engine with an electric generator, significantly enhancing both flight endurance time and payload capability for multi-rotor UAVs [24]. Beyond power systems, intelligent navigation technologies have increasingly incorporated artificial intelligence, 5G communications, and the BeiDou Navigation Satellite System to support autonomous take-off and landing, accurate trajectory tracking, and real-time obstacle avoidance in complex environments [25–27].

Furthermore, recent studies have demonstrated the rapid expansion of unmanned aerial vehicle (UAV) applications across a wide range of civil domains. For example, Higinio et al. provided a comprehensive review of civil UAV applications, showing that UAVs have evolved from military platforms into versatile tools for transportation, infrastructure inspection, environmental monitoring, disaster management, and precision agriculture [28]. Building upon this overview, Outay et al. specifically examined UAV applications in road safety and highway infrastructure management [29]. In the field of public security and intelligent surveillance, Zhou et al. proposed a UAV patrol system that combines panoramic image stitching with object detection to enhance large-scale scene monitoring [30]. To address more dynamic surveillance scenarios, Wang et al. introduced a human–UAV interactive perception framework based on a skeleton-guided iterative perspective optimization algorithm for tracking abnormal pedestrian behaviors [31]. Although these studies substantially improve perception capability, they mainly concentrate on surveillance performance and pay less attention to system-level robustness and multi-task coordination.

For disaster management, Qadir et al. systematically reviewed UAV-assisted disaster management in smart cities, emphasizing the integration of path planning algorithms with 5G communication technologies to enable rapid situational awareness, resource allocation, and emergency response [32]. Similarly, Yucesoy et al. summarized operations research applications of UAVs in disaster response, demonstrating that optimization techniques can significantly improve mission scheduling, search-and-rescue operations, and humanitarian logistics [33]. Besides, Palmeri et al. demonstrated the effectiveness of UAV-based remote sensing for monitoring the temporal evolution of geomorphological processes, providing high-resolution spatial data for environmental analysis [34]. In precision agriculture, Qiao et al. investigated the aerodynamic effects generated by multi-rotor UAVs on fruit-tree frost protection, showing that UAV-induced airflow can effectively mitigate frost damage under specific environmental conditions [35]. In medical logistics, Ong et al.

validated the feasibility of UAV transportation of frozen blood samples using phase change materials, demonstrating that UAV delivery can maintain sample quality while significantly reducing transportation time [36]. Collectively, these studies confirm the versatility of UAV technologies but also reveal that existing solutions are largely application-specific, motivating the development of more intelligent and adaptive UAV frameworks capable of operating reliably in complex real-world environments.

This study focuses on the instant delivery of meals using UAVs in the food delivery industry, a model that has seen numerous practical applications in China. In this domain, Liu proposed a mixed-integer programming model and developed an optimization-based progressive algorithm for online fleet scheduling optimization in UAV-based on-demand meal delivery [37]. Pinto et al. considered the quantity and location selection of charging stations in UAV meal delivery, using multi-objective optimization and a hyper-heuristic approach based on genetic algorithms and an improved particle swarm optimization algorithm [38]. Hamid et al. optimized the delivery operations of a homemade meal startup by integrating UAVs and crowdsourcing into its existing system [39]. Hana and Kob employed a mixed-integer linear programming method to minimize the total delivery completion time in a multi-restaurant environment using a shared UAV fleet [40]. Lu et al. further studied a multi-distribution-center drone–rider mode using order-region clustering and improved tabu search, with delivery cost and customer satisfaction as the principal criteria [41]. In comparison, the present study optimizes cost, makespan, emissions, and average customer tardiness simultaneously, explicitly searches a Pareto set, and assigns restaurant pickups and customer deliveries through candidate rendezvous points.

Taken together, the literature provides valuable models for UAV routing, hybrid ground–air delivery, and meal-delivery operations, but these streams are usually studied separately. In contrast to traditional single-fleet or single-objective studies, the proposed framework coordinates the two transport modes through candidate rendezvous points and generates a Pareto set from which platform operators can select plans under different cost, time, environmental, and punctuality priorities. The model remains a static planning formulation; dynamic orders and stochastic operating conditions are reserved for future work. Furthermore, in response to green and low-carbon development requirements, the formulation explicitly includes an emission objective. An improved NSGA-II integrating hybrid initialization, adaptive genetic operators, local search, structural repair, and archive-assisted diversity preservation is then used to solve the resulting four-objective problem. In summary, our research extends the state of the art as follows:

(1) We propose a collaborative UAV–rider meal delivery model and formulate a four-objective optimization problem considering operational cost, completion time, carbon emissions, and average customer tardiness.
(2) We develop an E-NSGA-II algorithm for the proposed model, integrating hybrid initialization, adaptive operators, local search, and repair/diversity mechanisms to improve feasibility, convergence, and Pareto-front distribution.
(3) We validate the proposed method through convergence, multi-algorithm comparison, scalability, ablation, operational sensitivity, and case-study analyses. Comparisons with NSGA-II, MOPSO, NSGA-III, MOEA/D, and discrete multi-objective adaptations of MODE and IMODE evaluate convergence, solution-set distribution, and computational cost under common experimental budgets.

The remainder of this paper is organized as follows: Section 2 describes the problem and presents the mathematical model. Section 3 details the design of the E-NSGA-II. Section 4 discusses the experimental setup, results, and analysis. Finally, Section 5 concludes the paper and suggests future research directions.

2. Problem Description and Model Construction

2.1. Problem Definition

In the traditional food delivery model, delivery riders receive orders before heading to restaurants to pick up meals and deliver them to customers. However, food delivery frequently coincides with peak meal times, leading to delays due to insufficient riders, traffic congestion, and other issues. Additionally, the urgency to deliver on time may result in unsafe practices, while the limitations on delivery range can affect the diversity of choices available to customers. Therefore, effectively shortening delivery times using faster delivery methods, such as drones, to assist riders in expediting the delivery process and reducing delivery costs has become a powerful strategy for enhancing competitiveness in the food delivery industry. In the context of national advocacy for energy conservation and emissions reduction, responsible and accountable companies must also consider their corporate social responsibility in relation to carbon emissions.

Addressing the aforementioned issues in traditional food delivery models, this paper proposes a drone-based food delivery route planning model that incorporates time windows. The model leverages the advantages of drones, such as high flight speed and immunity to complex ground traffic conditions. Furthermore, the use of drones can significantly reduce the number of riders, thereby lowering labor costs and simultaneously alleviating environmental pressure by decreasing reliance on ground transportation. Additionally, considering the complex residential environments in China, this paper optimizes the drone-based food delivery process as follows. After an order is received and the meal is prepared, a UAV departs from the distribution center to pick up the meal from the restaurant and transports it to a designated rendezvous point. The meal is then handed over to a delivery rider, who completes the last-mile delivery to the customer. The workflow of this integrated food delivery model is illustrated in Figure 1.



Figure 1. Schematic illustration of the collaborative UAV–rider meal delivery system.

Considering the constraints in real-world delivery environments, the following assumptions are made for this model:

(1) Each restaurant prepares one order at a time; preparation times are known and deterministic.
(2) UAV routes are represented as executable restaurant–rendezvous pickup–transfer pairs.
(3) Each package is transferred exactly once at a rendezvous point.

(4) Riders can serve multiple customers from a single rendezvous point.
(5) Customer delivery time windows are treated as soft service targets, with late arrivals penalized.
(6) UAVs and riders have limited capacities.
(7) Battery consumption is proportional to travel distance with payload.
(8) It is assumed that the drone will not require any additional waiting time when picking up meals at the restaurant, meaning it can depart immediately upon arrival.

This model aims not only to improve efficiency in food delivery but also to reflect the commitment of enterprises to sustainable development through reduced carbon emissions.

2.2. Mathematical Model

The Collaborative UAV–Rider Delivery Routing Problem (CUDRP) is defined on a multimodal transportation network. We formulate this as a mixed-integer linear programming (MILP) optimization problem. For clarity, the complete decision and auxiliary variables used in the formulation are consolidated in the appendix; the full sets, parameters, and notation are retained in the Abbreviations table.

Let the complete decision vector be

$$\boldsymbol{X} = \left(\boldsymbol{x}^U, \boldsymbol{x}^R, \boldsymbol{y}^U, \boldsymbol{z}^R, \boldsymbol{w}^R, \boldsymbol{\delta}^U, \boldsymbol{\delta}^R, \boldsymbol{\delta}^{RT}, \boldsymbol{a}^U, \boldsymbol{a}^R, \boldsymbol{d}^U, \boldsymbol{d}^R, \boldsymbol{L}^U, \boldsymbol{L}^R, \boldsymbol{\xi}, \boldsymbol{C}^U, \boldsymbol{C}^R, T_{\max}\right), \quad (1)$$

where the bold symbols collect the corresponding indexed variables defined in the Abbreviations table. Denote by Ω the feasible decision set satisfying the routing, capacity, timing, UAV endurance time, and transfer-consistency constraints. The four-objective problem is compactly written as

$$\min_{\boldsymbol{X}\in\Omega} \boldsymbol{F}(\boldsymbol{X}) = \min_{\boldsymbol{X}\in\Omega}(f_1(\boldsymbol{X}), f_2(\boldsymbol{X}), f_3(\boldsymbol{X}), f_4(\boldsymbol{X})). \quad (2)$$

2.2.1. Objective Functions

(1) Minimize total operational cost:

$$\min\ f_1 = \sum_{k\in K^U}\sum_{i\in V}\sum_{j\in V} c_{ij}^U x_{ijk}^U + \sum_{k\in K^R}\sum_{i\in V}\sum_{j\in V} c_{ij}^R x_{ijk}^R + FC_U \cdot \sum_{k\in K^U} \delta_k^U + FC_R \cdot \sum_{k\in K^R} \delta_k^R \quad (3)$$

This objective captures variable costs (distance-based) and fixed costs (vehicle activation). The first two terms represent operational costs proportional to travel distance, while the last two terms account for fixed costs associated with deploying UAVs and riders.

(2) Minimize maximum completion time:

$$\min\ f_2 = T_{\max} \quad (4)$$

subject to

$$T_{\max} \geq C_k^U \quad \forall k \in K^U \quad (5)$$

$$T_{\max} \geq C_k^R \quad \forall k \in K^R \quad (6)$$

$$C_k^U \geq d_{ik}^U - \mathcal{M}\left(1 - \sum_{j\in V} x_{ijk}^U\right) \quad \forall i \in V, \forall k \in K^U \quad (7)$$

$$C_k^R \geq d_{ik}^R - \mathcal{M}\left(1 - \sum_{j\in V} x_{ijk}^R\right) \quad \forall i \in V, \forall k \in K^R \quad (8)$$

This objective minimizes the overall completion time of the delivery operation. C_k^U denotes the completion time of UAV k after finishing its pickup–transfer route, while C_k^R

denotes the completion time of rider k after completing the assigned customer delivery route. The auxiliary variable $T_{\max}$ linearizes the maximum completion time over both fleets.

(3) Minimize total carbon emissions:

$$\min\ f_3 = \sum_{k\in K^U}\sum_{i\in V}\sum_{j\in V} em_{ij}^U x_{ijk}^U + \sum_{k\in K^R}\sum_{i\in V}\sum_{j\in V} em_{ij}^R x_{ijk}^R \tag{9}$$

This environmental objective quantifies the total carbon footprint of the delivery operation. Emissions are assumed to be proportional to distance traveled, with different coefficients for UAVs and riders based on their energy consumption profiles.

(4) Minimize average customer tardiness:

$$\min\ f_4 = \frac{1}{|V_C|}\sum_{c\in V_C}\xi_c \tag{10}$$

This soft objective measures the mean late-arrival time across all customers, in minutes per customer. Consistent with the computational implementation, customer time windows are service targets: late arrivals do not increase the hard constraint-violation measure, but contribute through the tardiness slack ξ_c. Early arrivals wait until the opening time, whereas arrivals later than l_c remain allowable.

2.2.2. Constraints

The following constraints prohibit self-loops in UAV and rider routes, ensuring that a node cannot be counted as visited through an arc from itself to itself:

$$x_{iik}^U = 0 \quad \forall i \in V, \forall k \in K^U \tag{11}$$

$$x_{iik}^R = 0 \quad \forall i \in V, \forall k \in K^R \tag{12}$$

(1) UAV routing constraints:

$$\sum_{k\in K^U}\sum_{j\in V} x_{rjk}^U = 1 \quad \forall r \in V_R \tag{13}$$

$$\sum_{j\in V} x_{0jk}^U \le 1 \quad \forall k \in K^U \tag{14}$$

$$\sum_{i\in V} x_{ihk}^U - \sum_{j\in V} x_{hjk}^U = 0 \quad \forall h \in V_R \cup V_T,\ \forall k \in K^U \tag{15}$$

$$\sum_{i\in V} x_{itk}^U \le 1 \quad \forall t \in V_T,\ \forall k \in K^U \tag{16}$$

$$L_{0k}^U = 0 \quad \forall k \in K^U \tag{17}$$

$$L_{jk}^U \ge L_{ik}^U + d_j - \mathcal{M}(1 - x_{ijk}^U) \quad \forall i, j \in V,\ \forall k \in K^U \tag{18}$$

$$L_{ik}^U \le Q_U \quad \forall i \in V,\ \forall k \in K^U \tag{19}$$

Constraint (13) ensures each restaurant is visited exactly once. Constraint (14) limits each UAV to at most one route starting from the depot. Constraint (15) maintains flow conservation at intermediate nodes. Constraint (16) restricts each UAV to visit each rendezvous point at most once. Constraint (17)–(19) track and limit payload capacity.

(2) Rider routing constraints:

$$\sum_{k\in K^R}\sum_{j\in V} x_{cjk}^R = 1 \quad \forall c \in V_C \tag{20}$$

$$\sum_{c\in V_C} w_{ctk}^R \le |V_C|\delta_{tk}^R \quad \forall t \in V_T,\ \forall k \in K^R \tag{21}$$

$$\delta_{tk}^{R} \le \delta_{k}^{R} \quad \forall t \in V_T,\ \forall k \in K^R \tag{22}$$

$$\sum_{i \in V} x_{ihk}^{R} - \sum_{j \in V} x_{hjk}^{R} = 0 \quad \forall h \in V_C,\ \forall k \in K^R \tag{23}$$

$$L_{tk}^{R} = \sum_{c \in V_C} d_c w_{ctk}^{R} \quad \forall t \in V_T,\ \forall k \in K^R \tag{24}$$

$$L_{tk}^{R} \le Q_R \quad \forall t \in V_T,\ \forall k \in K^R \tag{25}$$

$$L_{ik}^{R} \le Q_R \quad \forall i \in V,\ \forall k \in K^R \tag{26}$$

$$w_{ctk}^{R} \le z_{ct}^{R} \quad \forall c \in V_C, \forall t \in V_T, \forall k \in K^R \tag{27}$$

$$w_{ctk}^{R} \le \delta_{tk}^{R} \quad \forall c \in V_C, \forall t \in V_T, \forall k \in K^R \tag{28}$$

$$\sum_{t \in V_T} \sum_{k \in K^R} w_{ctk}^{R} = 1 \quad \forall c \in V_C \tag{29}$$

Constraint (20) ensures each customer has one outgoing rider arc. Constraints (21) and (22) activate rider-rendezvous and rider-use variables only when customers are served from that point. Constraint (23) maintains rider flow conservation at customer nodes. Constraints (24)-(26) track and limit rider load, while constraints (27)–(29) link customer-rendezvous assignments, rider assignments, and customer service decisions without using products of binary variables.

(3) Time and service-penalty constraints

$$\sum_{j \in V} x_{0jk}^{U} = \delta_{k}^{U} \quad \forall k \in K^U \tag{30}$$

$$a_{rk}^{U} \ge q_r^{p} - \mathcal{M}\left(1 - \sum_{i \in V} x_{irk}^{U}\right) \quad \forall r \in V_R, \forall k \in K^U \tag{31}$$

$$a_{ck}^{R} \ge e_c - \mathcal{M}\left(1 - \sum_{i \in V} x_{ick}^{R}\right) \quad \forall c \in V_C, \forall k \in K^R \tag{32}$$

$$\xi_c \ge a_{ck}^{R} - l_c - \mathcal{M}\left(1 - \sum_{i \in V} x_{ick}^{R}\right) \quad \forall c \in V_C, \forall k \in K^R \tag{33}$$

$$\xi_c \ge 0 \quad \forall c \in V_C \tag{34}$$

$$\sum_{i \in V} \sum_{j \in V} t_{ij}^{U} x_{ijk}^{U} \le E_U \quad \forall k \in K^U \tag{35}$$

Constraint (30) links UAV route departure to the UAV activation variable. Constraint (31) guarantees that a UAV serving restaurant r respects the food preparation time. Constraint (32) represents the lower bound of the customer time window for the rider that visits customer c, while constraints (33) and (34) define the non-negative tardiness slack used in objective f_4. Thus, the latest delivery time is modeled as a soft service target rather than as a hard infeasibility condition. Constraint (35) limits UAV flight time according to the UAV endurance time budget.

(4) Transfer consistency constraints:

$$\sum_{t \in V_T} y_{rt}^{U} = 1 \quad \forall r \in V_R \tag{36}$$

$$\sum_{t \in V_T} z_{ct}^{R} = 1 \quad \forall c \in V_C \tag{37}$$

$$y_{rt}^{U} \leq \sum_{k \in K^{U}} x_{rtk}^{U} \quad \forall r \in V_R, \forall t \in V_T \tag{38}$$

$$\sum_{k \in K^{R}} w_{ctk}^{R} = z_{ct}^{R} \quad \forall c \in V_C, \forall t \in V_T \tag{39}$$

These constraints ensure each restaurant pickup is assigned to exactly one transfer point (36), each customer delivery is associated with exactly one transfer point (37), and these assignments are consistent with the actual routes taken by UAVs and riders (38) and (39). In particular, constraint (38) enforces direct restaurant-to-rendezvous consistency in the current implementation; a more general formulation that allows several restaurants to be visited before a later rendezvous point would require additional route-membership and visit-order variables.

The proposed model extends traditional vehicle routing problems in several significant ways: (1) restaurant pickup, aerial transfer, and ground delivery are modeled as distinct coordinated phases; (2) transfer-consistency constraints link restaurant pickups, rendezvous assignments, and rider routes; (3) each restaurant pickup and customer delivery is coordinated through rendezvous assignments, although order-level commodity flows between specific restaurant–customer pairs are not explicitly tracked; and (4) preparation times and soft customer windows introduce temporal dependencies, with service punctuality measured by average customer tardiness. The CUDRP is NP-hard as it generalizes the Vehicle Routing Problem with Time Windows (VRPTW) and the Pickup and Delivery Problem. The number of binary variables is $O((K_U + K_R) \cdot |V|^2 + M \cdot P + N \cdot P)$, making exact solution methods impractical for realistically sized instances.

3. Enhanced NSGA-II Algorithm Design

3.1. Overall Algorithm Framework

The Enhanced NSGA-II (E-NSGA-II) algorithm follows the general NSGA-II framework but incorporates several problem-specific enhancements. The overall procedure is given in Algorithm 1. In the pseudocode, $\mathrm{CV}(s)$ denotes the residual hard constraint-violation value of solution s, while average customer tardiness is evaluated through the fourth objective rather than through $\mathrm{CV}(s)$.

NSGA-II is selected as the base framework because it directly handles conflicting objectives without requiring predefined aggregation weights, it retains multiple non-dominated alternatives through Pareto ranking, and it supports permutation- and assignment-based operators for the mixed route representation. Each enhancement addresses a specific search difficulty: hybrid initialization improves initial solution quality and coverage; adaptive variation balances exploration and exploitation over generations; local search refines promising route structures; deterministic repair restores chromosome legality after variation; and the bounded archive with diversity replenishment preserves sparsely represented regions of the approximation front. The ablation study and expanded multi-algorithm comparison in Section 4.6 and Section 4.4, respectively, evaluate the resulting quality–runtime trade-offs.

3.2. Solution Representation and Decoding Procedure

A chromosome in E-NSGA-II consists of four parts, as illustrated in Figure 2:

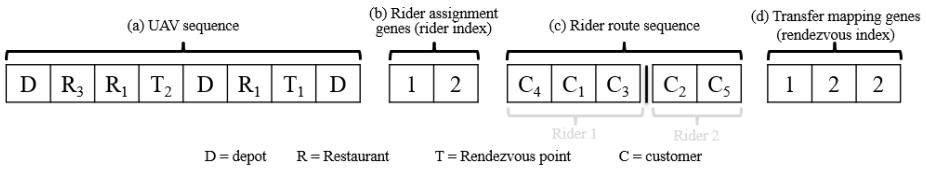


Figure 2. Chromosome representation for the collaborative UAV–rider delivery problem.

Algorithm 1 Enhanced NSGA-II (E-NSGA-II) Framework

Require: Instance $\mathcal{I}$, population size N, generations $G_{\max}$, archive limit N_A, local-search parameters (p_{ls}, ρ_{ls})
Ensure: Final non-dominated solution set $\mathcal{P}^*$
$P \leftarrow \textsc{HybridInitialization}(\mathcal{I}, N)$ ▷ Algorithm 2
$\textsc{DecodeAndEvaluate}(P)$; $A \leftarrow \emptyset$
for $g = 0$ **to** $G_{\max} - 1$ **do**
  Compute $p_c \leftarrow p_c(g)$ and $p_m \leftarrow p_m(g)$ using Equations (40)–(41)
  $\textsc{RankAndCrowdingAssignment}(P)$
  $M \leftarrow \textsc{BinaryTournamentSelection}(P, N)$
  $Q \leftarrow \emptyset$
  for all parent pairs (P_1, P_2) in M **do**
    $(O_1, O_2) \leftarrow \textsc{AdaptiveVariation}(P_1, P_2, p_c, p_m)$ ▷ Algorithm 3
    for all $O \in \{O_1, O_2\}$ **do**
      repair missing/duplicate restaurants and restaurant–rendezvous pairs
      repair customer duplication, omission, and rider-route assignment
      $\textsc{DecodeAndEvaluate}(O)$
    end for
    if $g > \rho_{ls} G_{\max}$ **and** $\text{rand}() < p_{ls}$ **then**
      $O_1 \leftarrow \textsc{LocalSearch}(O_1)$; $O_2 \leftarrow \textsc{LocalSearch}(O_2)$ ▷ Algorithm 4
    end if
    $Q \leftarrow Q \cup \{O_1, O_2\}$
  end for
  $P \leftarrow \textsc{EnvironmentalSelection}(P \cup Q, N)$
  $P \leftarrow \textsc{DiversityReplenishment}(P, g)$
  $A \leftarrow \textsc{UpdateNondominatedArchive}(A \cup P, N_A)$
  if $(g+1) \bmod 10 = 0$ **then**
    $P \leftarrow \textsc{InjectSparseArchiveElites}(P, A, \lceil 0.05N \rceil)$
  end if
end for
return $\mathcal{P}^* \leftarrow \textsc{FirstNondominatedFront}(P)$

The chromosome consists of four code-level parts. Part 1 stores the UAV route sequence, where depot markers (0) separate UAV routes and each restaurant is followed by its assigned rendezvous point. Part 2 is an integer vector indexed by rendezvous points, and each gene gives the assigned rider index; for example, $[1, 2]$ under $(T1, T2)$ means $T1 \rightarrow$ Rider 1 and $T2 \rightarrow$ Rider 2 after decoding. Part 3 stores capacity-bounded customer-tour segments grouped by rider, while each tour origin is inherited from the rendezvous assignment in Part 2 rather than independently encoded. If the same rider index appears in more than one segment then the decoder treats them as consecutive tours: the rider returns to the assigned rendezvous between segments, and elapsed time is propagated rather than reset. Part 4 is an integer vector indexed by restaurants and specifies the rendezvous index for each pickup; for example, $[1, 2, 2]$ under $(R1, R2, R3)$ decodes as $R1 \rightarrow T1$, $R2 \rightarrow T2$, and $R3 \rightarrow T2$.

The decoder deterministically converts the chromosome into a complete delivery plan by parsing depot-separated UAV routes, enforcing restaurant–rendezvous mappings, attaching assigned rendezvous origins to rider customer sequences, and evaluating the resulting schedule. Route executability and UAV endurance time are treated as hard feasibility checks, whereas customer-window performance is evaluated by the average tardiness per customer. The current implementation coordinates UAV–rider hand-off topology through rendezvous assignments, but it does not explicitly track order-level commodity flows or propagate UAV-side rendezvous arrival times into rider departure times as a separate temporal synchronization constraint.

3.3. Hybrid Initialization Strategy

To ensure both diversity and solution quality in the initial population, we employ a hybrid initialization strategy that combines greedy construction and randomized construction. The greedy part accounts for approximately 25% of the population and uses nearest-rendezvous assignment, angular partitioning of restaurants, nearest-neighbor sequencing, and savings-based rider route construction. The remaining individuals are generated by randomized transfer assignment, randomized UAV route partitioning, and capacity-aware customer splitting. The procedure is specified in Algorithm 2.

Algorithm 2 Hybrid Initialization

Require: Instance data $\mathcal{I}$, population size N
Ensure: Initial population P_0
$P_0 \leftarrow \emptyset$
$N_g \leftarrow \max\{1, \lfloor 0.25N \rfloor\}$
for $h = 1$ **to** N_g **do**
    assign every restaurant to its nearest rendezvous point
    partition restaurants angularly and order each UAV route by nearest neighbor
    construct capacity-aware rider routes using the savings heuristic
    $s \leftarrow$ EncodeCurrentAssignments($\mathcal{I}$)
    DecodeAndEvaluate(s)
    $P_0 \leftarrow P_0 \cup \{s\}$
end for
while $|P_0| < N$ **do**
    randomly sample transfer mappings and depot-separated UAV routes
    randomly permute customers and split rider routes by capacity and route length
    $s \leftarrow$ EncodeCurrentAssignments($\mathcal{I}$)
    DecodeAndEvaluate(s)
    $P_0 \leftarrow P_0 \cup \{s\}$
end while
return P_0

3.4. Adaptive Genetic Operators

We employ generation-dependent adaptive probabilities for crossover and mutation, as shown in Algorithm 3. The implementation gradually decreases the crossover probability and increases the mutation probability as the search progresses:

$$p_c(g) = p_c^{max} - (p_c^{max} - p_c^{min}) \frac{g}{G_{\max} - 1} \tag{40}$$

$$p_m(g) = p_m^{min} + (p_m^{max} - p_m^{min}) \frac{g}{G_{\max} - 1} \tag{41}$$

where $g = 0, \ldots, G_{\max} - 1$ is the current generation. This schedule encourages stronger recombination in the early search stage and gradually increases mutation-based exploration in later generations.

The crossover operator extracts restaurant permutations from UAV sequences, applies order crossover to the restaurant order, inherits UAV route sizes from the parents, and rebuilds executable restaurant–rendezvous pairs using a uniform transfer-mapping crossover. The mutation stage contains four implemented moves: inter-restaurant swap, inter-route restaurant relocation, restaurant–rendezvous remapping, and intra-rider customer swap. The corresponding pseudocode is given in Algorithm 3.

Algorithm 3 Adaptive Crossover and Mutation

Require: Parent chromosomes P_1 and P_2, crossover probability p_c, mutation probability p_m
Ensure: Offspring chromosomes O_1 and O_2
$O_1 \leftarrow \text{CLONE}(P_1)$; $O_2 \leftarrow \text{CLONE}(P_2)$
if rand() $< p_c$ **then**
  apply order crossover to restaurant permutations
  inherit route sizes and uniformly recombine transfer mappings
  rebuild restaurant–rendezvous pairs in (O_1, O_2)
end if
for all $O \in \{O_1, O_2\}$ **do**
  if rand() $< p_m$ **then**
    sample from restaurant swap, inter-route relocation, rendezvous remapping, and intra-rider customer swap
  end if
end for
return (O_1, O_2)

3.5. Local Search Mechanism

Local search is applied as a late-stage intensification mechanism. In the implementation, it is enabled only after 75% of the maximum generations have elapsed, and each newly generated offspring pair is selected for local search with probability 0.05. The local search uses two bounded neighborhoods: a first-improvement reversal of restaurant order within one UAV route, followed by reconstruction of its restaurant–rendezvous pairs, and an inter-route customer relocation move on rider routes. A candidate move is accepted only when the decoded neighbor is hard-feasible and improves the operational-cost objective used by the local refinement routine. The process is specified in Algorithm 4.

Algorithm 4 Problem-Specific Local Search

Require: Solution S, UAV reversal trial limit L_{2opt}, rider-relocation trial limit L_{reloc}, and acceptance limit A_{reloc}
Ensure: Locally improved solution S'
$S' \leftarrow \text{CLONE}(S)$
$S' \leftarrow \text{FIRSTIMPROVINGUAV2OPT}(S', L_{2opt})$
$S' \leftarrow \text{BOUNDEDFIRSTIMPROVINGRIDERRELOCATION}(S', L_{reloc}, A_{reloc})$
return S'

3.6. Constraint Handling and Repair Mechanism

Effective constraint handling is important for solving the CUDRP because the search space is constrained by routing structure, capacity, temporal dependencies, transfer consistency, and UAV endurance time limits. A hybrid repair–penalty mechanism therefore corrects common structural violations while retaining objective-based treatment of soft service targets.

The repair mechanism is invoked after variation and follows the same deterministic sequence used in Algorithm 1. First, the UAV chromosome is parsed at depot separators; duplicate restaurant genes are removed, omitted restaurants are inserted into the shortest available restaurant route, and every restaurant is followed by the rendezvous encoded in the transfer-mapping vector. Second, rider tours are scanned to remove duplicate or invalid customer genes, omitted customers are distributed across the retained tours, and rendezvous-to-rider assignments are rebuilt for active riders. Consecutive tours with the same rider index are linked by a return to the assigned rendezvous during decoding. The repaired chromosome is then decoded. Average customer tardiness remains in f_4,

whereas residual UAV endurance time or per-tour rider-capacity excess contributes to $\mathrm{CV}(s)$ and is handled by constraint-dominance during selection and archive maintenance.

Solutions that remain partially infeasible after repair are retained in the population but compared using a constraint-violation measure before objective dominance is considered. This measure covers residual hard violations such as UAV endurance time, whereas average customer tardiness is evaluated directly through f_4. This design avoids imposing a hard latest-arrival constraint and optimizing the same violation as a soft objective.

3.7. Diversity Preservation

Maintaining population diversity is essential for preventing premature convergence and obtaining a well-distributed Pareto front. In E-NSGA-II, diversity preservation is enhanced through three complementary mechanisms.

First, the standard crowding distance of NSGA-II is retained to maintain diversity in objective space. In addition, repeated objective-vector signatures are monitored after environmental selection. When their proportion exceeds the prescribed threshold in the second half of the run, low-priority individuals are replaced by newly generated greedy or random chromosomes, thereby reopening under-explored routing and transfer-assignment regions.

Second, an external archive stores distinct hard-feasible non-dominated solutions discovered during evolution. Duplicate chromosome/objective signatures are removed, and when the archive exceeds 50 solutions crowding distance is used to retain boundary and sparsely represented solutions. Every 10 generations, up to 5% of the population is replaced by sparse archive elites that are not already present; the final reported approximation remains the last population's first non-dominated front, consistent with the benchmark evaluation protocol.

Together, crowding-based environmental selection, duplicate-triggered replenishment, and the bounded external archive balance convergence with solution-set coverage while keeping the archive-management cost controlled.

4. Computational Experiments and Analysis

4.1. Experimental Design and Setup

The effectiveness of the proposed E-NSGA-II algorithm was evaluated through several complementary experiments. All the experiments were implemented in Python 3.9, and the figures were regenerated using publication-oriented plotting styles based on SciencePlots/Matplotlib.

Benchmark instances were generated using the same clustered-random instance generator embedded in the simulation code. Small-scale instances contained 10–20 customers, 3–5 restaurants, and 2–3 rendezvous points; medium-scale instances contained 25–40 customers, 6–8 restaurants, and 3–5 rendezvous points; and large-scale instances contained 50–100 customers, 8–12 restaurants, and 5–8 rendezvous points. Customer and restaurant locations followed clustered urban spatial patterns, while rendezvous points were placed near central or demand-sensitive regions to support UAV–rider hand-off operations. Key operational parameters were fixed as UAV speed 40 km/h, rider speed 15 km/h, UAV capacity 5 kg, and rider capacity 20 kg. The scale-dependent instance settings and E-NSGA-II algorithm parameters are summarized in Table 1.

All the compared algorithms shared the same chromosome representation, decoder, and objective evaluation.

Table 1. Algorithm parameters.

Scale	Customer Nodes	Restaurant Nodes	Rendezvous	Radius	Time Window	Pop. Size	Max. Gen.
Small	10–20	3–5	2–3	0.5–2.0 km	30–60 min	50	100
Medium	25–40	6–8	3–5	1.5–3.0 km	40–75 min	100	200
Large	50–100	8–12	5–8	2.0–5.0 km	45–90 min	120	300

Scale-independent parameters: $p_c^{min} = 0.70$, $p_c^{max} = 0.92$, $p_m^{min} = 0.08$, $p_m^{max} = 0.20$. local search probability = 0.05, local search start generation ratio = 0.75, UAV 2-opt maximum trials $L_{2opt} = 60$, and maximum archive size = 50.

4.2. Case Study: Pareto Analysis and Collaborative Delivery Visualization

To illustrate the decision structure returned by E-NSGA-II, a representative medium-scale scenario was solved in five independent runs with a population of 100 and 200 generations. Mining the saved histories and removing duplicates yielded 80 global non-dominated solutions. The normalized-ideal compromise solution is visualized in Figures 3–6, and its active decisions are reported in Table 2.

Figure 3 shows the modeled coordination topology. The UAV visits restaurants and their assigned rendezvous points, while riders perform capacity-bounded customer tours from the used rendezvous points. Green triangles denote the modeled transfer locations; the figure represents assignment consistency rather than separately validated real-time hand-off synchronization.

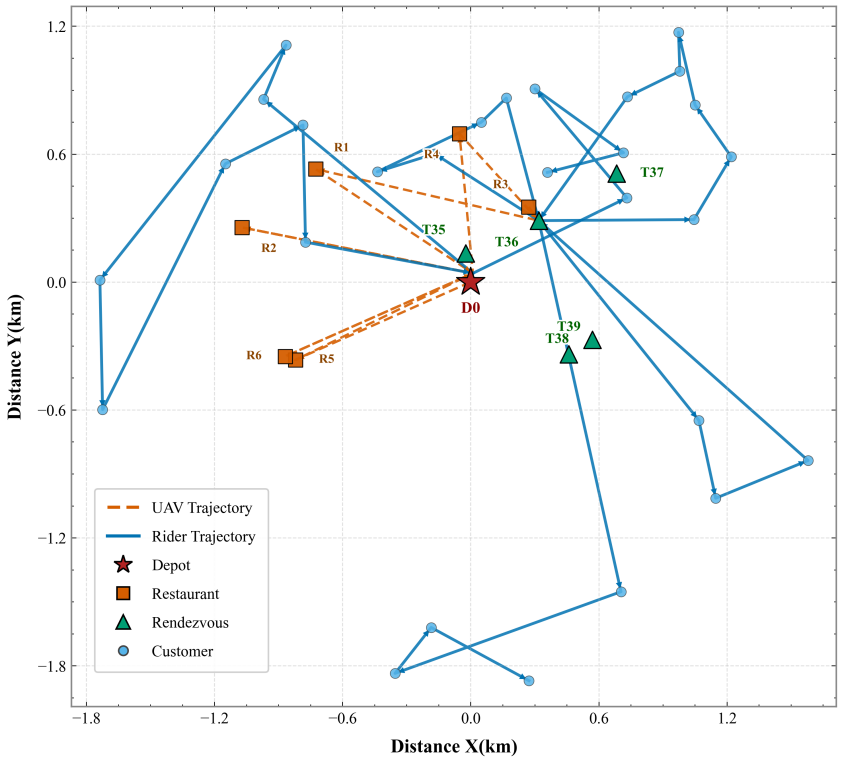


Figure 3. A representative collaborative UAV–rider routing topology generated by E-NSGA-II.

The selected plan used two rendezvous points, T35 and T36. Repeated tours assigned to the same rider were executed consecutively; every non-final tour returned to its assigned rendezvous before the next tour began, and the elapsed time was propagated across tours.

Figure 4 further illustrates the restaurant–rendezvous–rider–customer assignment relationships. Each restaurant is mapped to one transfer location, and each customer appears in exactly one rider tour. As noted in Section 3.2, these relationships do not constitute explicit order-level commodity-flow tracking between specific restaurant–customer pairs.

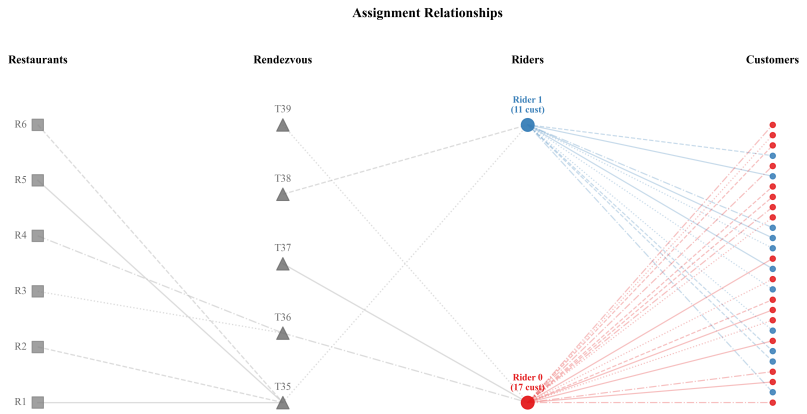


Figure 4. Multi-echelon assignment logic graph (Restaurant → Rendezvous → Rider → Customer) illustrating the rendezvous-based coordination structure of the selected solution.

Table 2. Active decision variables of the selected compromise solution.

Decision Group	Identifier	Active Value
UAV route arcs	UAV 1	D0 → R5 → T35 → R2 → T35 → R6 → T35 → R4 → T36 → R3 → T36 → R1 → T35 → D0
Restaurant–transfer mapping	Active y_{rt}^{U}	R1 → T35; R2 → T35; R3 → T36; R4 → T36; R5 → T35; R6 → T35
Used rendezvous–rider assignment	Active δ_{tk}^{RT}	T35 → Rider 1; T36 → Rider 0
Rider route arcs	Rider 0, tour 1	T36 → C26 → C9 → C32 → C15 → T36
Rider route arcs	Rider 0, tour 2	T36 → C7 → C13 → C28 → C19 → C34 → C21 → T36
Rider route arcs	Rider 0, tour 3	T36 → C17 → C25 → C30 → T36
Rider route arcs	Rider 0, tour 4	T36 → C16 → C27 → C33 → C10
Rider route arcs	Rider 1, tour 1	T35 → C23 → C12 → C18 → C24 → C20 → C11 → C14 → T35
Rider route arcs	Rider 1, tour 2	T35 → C8 → C29 → C31 → C22

To analyze the trade-offs among the objectives, Figure 5 presents the total operational cost, makespan, and average customer tardiness, while the color gradient represents carbon emissions. The obtained Pareto front demonstrates clear trade-off relationships among the objectives. Solutions with lower operational costs may lead to longer completion times or higher service penalties, whereas solutions emphasizing service quality may require higher operational costs or emissions. The Pareto distribution reveals structural conflicts between operational efficiency objectives and service-oriented objectives, indicating that improvements in delivery responsiveness may require additional operational and environmental costs.

The selected compromise solution, marked by the red star in Figure 5, exhibits relatively moderate values across the four objectives. The color gradient indicates the carbon emissions objective, the semi-transparent fitted surface highlights the overall shape of the Pareto manifold, and the red star marks the selected compromise solution mapped in Figures 3 and 4. These results suggest that E-NSGA-II is capable of generating alternative routing plans corresponding to different operational preferences.

Figure 6 presents a parallel-coordinate visualization of the normalized objective values for the obtained non-dominated solutions. The highlighted dark-red line solution exhibits relatively moderate objective values, indicating that the selected compromise solution avoids excessive optimization of any single objective at the expense of the others.

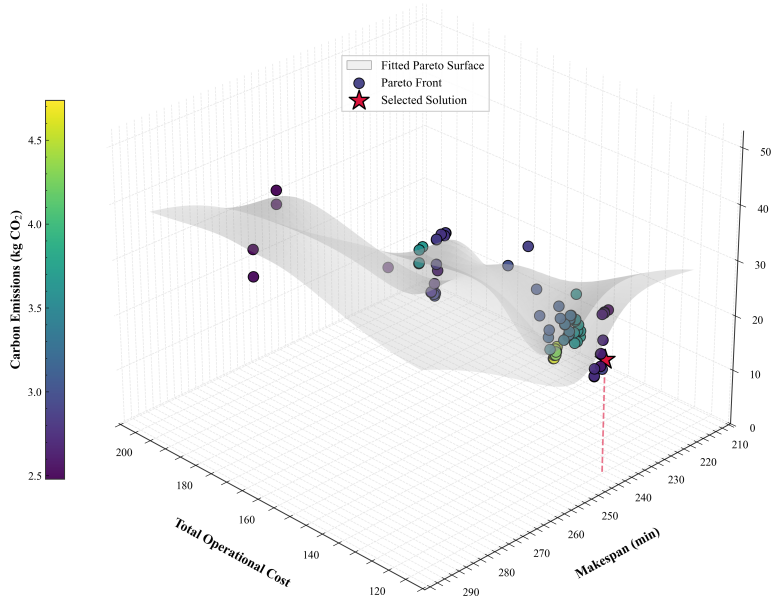


Figure 5. 3D Pareto trade-off visualization showing the relationship among total cost, makespan, and average customer tardiness.

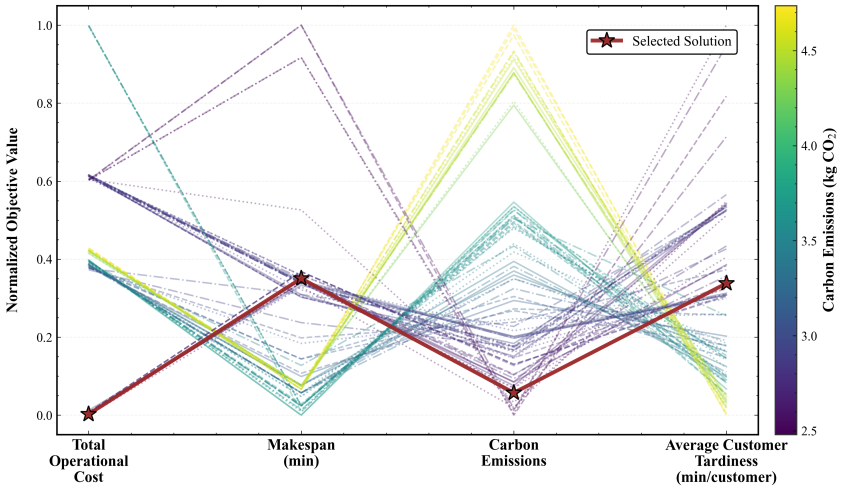


Figure 6. Parallel coordinate analysis of normalized objectives for the non-dominated solutions.

The case study illustrates that the proposed E-NSGA-II algorithm can generate feasible collaborative UAV–rider delivery plans under realistic urban delivery conditions. The obtained Pareto solutions illustrate the trade-offs among operational efficiency, service quality, environmental impact, and operational cost in UAV-assisted instant-delivery systems.

4.3. Sensitivity Analysis

4.3.1. Time-Window Width and Service Punctuality

Because customer time windows are soft service targets, this experiment evaluated service punctuality rather than route executability. Customer-window widths were varied from 40 to 120 min on Medium Instance. The indicator was the minimum average customer tardiness attained in the final Pareto set; the zero-tardiness solution ratio was also retained in the result data but remained zero under the tighter settings and was therefore less informative graphically.

As shown in Figure 7, service-punctuality performance generally improved as the window became wider, and minimum average customer tardiness showed a downward trend. For E-NSGA-II, the mean minimum value decreased from 11.258 min/customer at

40 min to 0.154 min/customer at 120 min. The corresponding values decreased from 13.809 to 0.671 for NSGA-II and from 23.717 to 2.298 for MOPSO. E-NSGA-II gave the lowest mean average tardiness at every tested width, although this does not imply that every solution was fully on time.

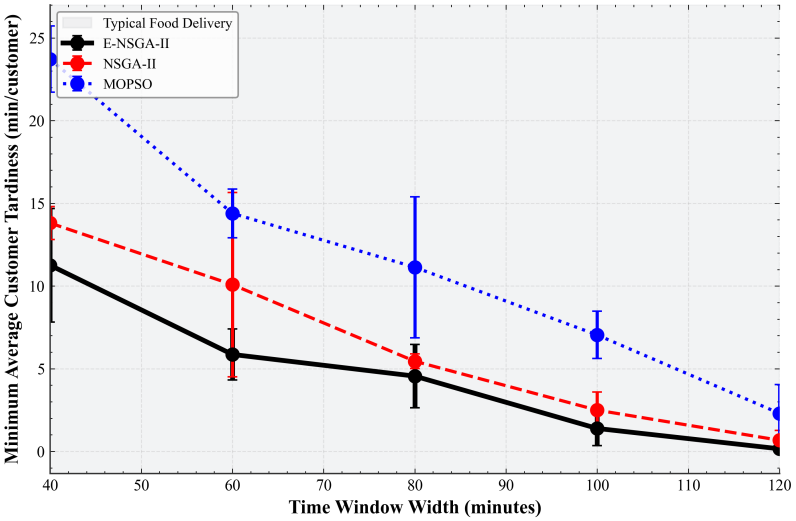


Figure 7. Sensitivity analysis of the minimum average customer tardiness for customer time-window widths from 40 to 120 min. Error bars show one standard deviation over three independent runs.

The result quantifies the service effect of a soft constraint: narrower windows do not make a plan mathematically infeasible, but they substantially increase average customer tardiness. Moderate relaxation provides more sequencing flexibility, whereas near-zero average tardiness appears only at the widest setting for some runs. The finding should therefore be interpreted as a service-level trade-off rather than as an executability measure.

4.3.2. UAV Endurance Time and Rider Speed

The operational-parameter sensitivity test used the same medium instance, a population size of 60 and 100 generations. The reported UAV endurance times were 40, 45, 50, and 60 min, while rider speed was varied over 10, 12.5, 15, 17.5, and 20 km/h. For each run, the minimum makespan and minimum average customer tardiness in the final Pareto set were extracted. This front-level reporting avoided making the sensitivity conclusion depend on a different subjective compromise solution at each parameter value.

Figure 8 shows that increasing the reported UAV endurance time from 40 to 60 min did not guarantee a monotonic makespan reduction. The mean minimum Pareto makespan remained between 208.49 and 211.23 min across the four settings, indicating that once sufficient route options are available, assignment and sequencing decisions dominate the remaining completion-time variation.

Increasing rider speed generally improved the attainable completion time. The mean minimum Pareto makespan decreased from 220.80 min at 10 km/h to 202.58 min at 20 km/h, with intermediate values of 217.91, 211.23, and 206.10 min. The minimum average customer tardiness also fell overall from 12.340 to 9.731 min/customer, although the 15 km/h setting exhibited a local increase because route reselection and stochastic search altered which trade-off entered the final front. Operationally, UAV endurance time limited the available route structures, whereas rider speed provided timing flexibility but could not replace coordinated assignment and sequencing. UAV endurance time here was an operational time budget, not a detailed electrochemical battery model.

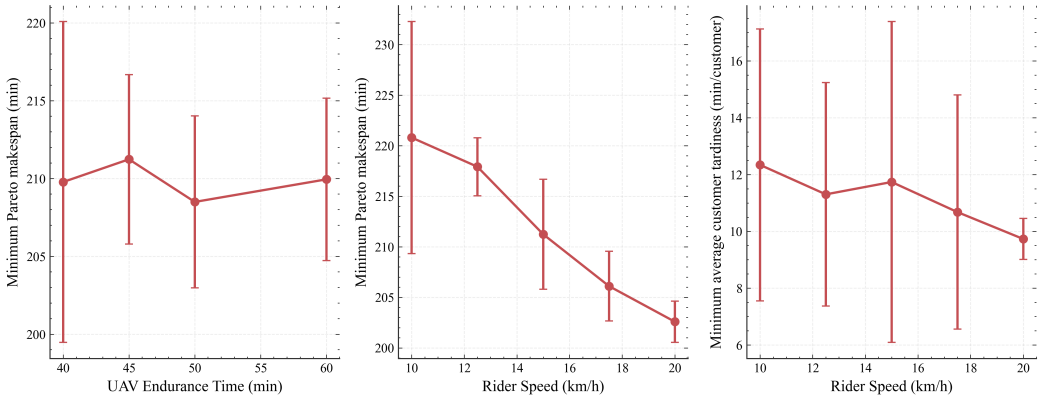


Figure 8. Sensitivity of attainable Pareto performance to UAV endurance time and rider speed. Error bars show one standard deviation over five independent runs.

4.4. Convergence and Diversity Analysis

To evaluate E-NSGA-II, comparative experiments were conducted against NSGA-II, MOPSO, NSGA-III, MOEA/D, MODE, and IMODE on a representative large-scale instance. NSGA-III used reference-direction niching, MOEA/D used Tchebycheff decomposition and neighborhood replacement, and the discrete MODE/IMODE adaptations implemented differential moves through position swaps and order-preserving crossover, with IMODE additionally using p-best guidance and success-history parameter updates. All the methods retained the common chromosome and objective evaluator, while the proposed enhancement modules were enabled only for E-NSGA-II. HV and IGD assessed front approximation and convergence, Spread evaluated distribution uniformity, and MNND was retained as an auxiliary nearest-neighbor-spacing indicator. Figure 9 and Table 3 both report all seven methods.

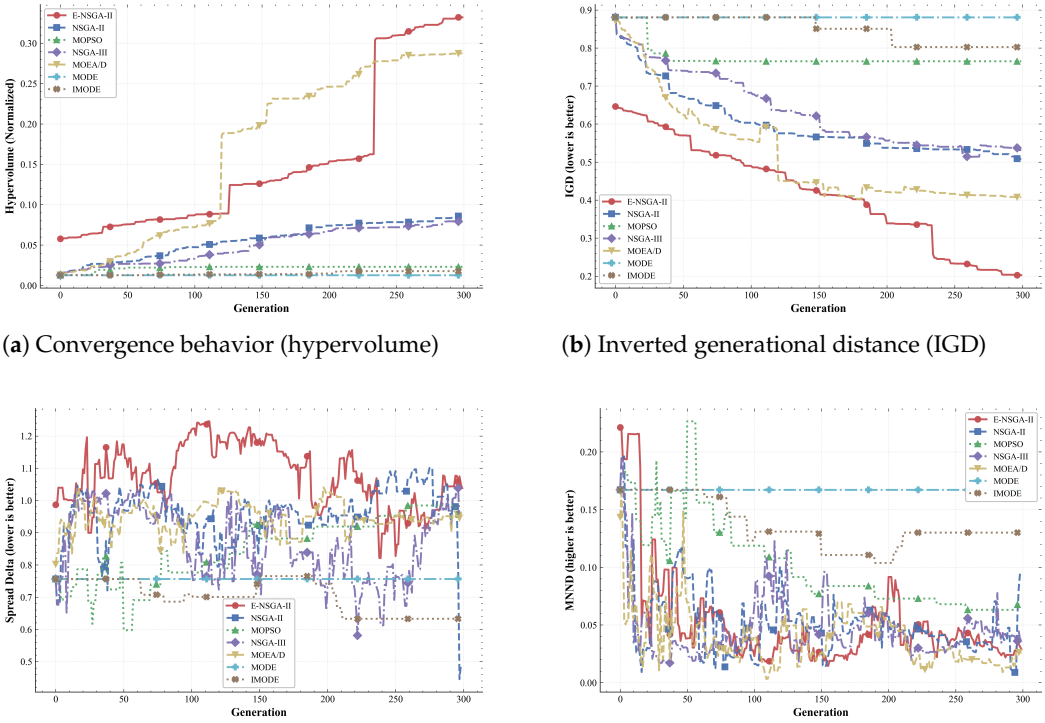
Table 3. Final performance over 10 independent runs on large instance (mean ± standard deviation for HV and IGD).

Algorithm	HV ↑	IGD ↓	Spread ↓	MNND	Runtime (s)
NSGA-II	0.031 ± 0.013	0.385 ± 0.064	0.955	0.036	1.238
MOPSO	0.015 ± 0.004	0.483 ± 0.047	0.894	0.082	**0.499**
NSGA-III	0.029 ± 0.008	0.394 ± 0.086	0.925	0.066	1.368
MOEA/D	0.044 ± 0.029	0.356 ± 0.051	0.919	0.154	1.037
MODE	0.011 ± 0.004	0.551 ± 0.083	**0.791**	0.170	1.343
IMODE	0.012 ± 0.004	0.539 ± 0.076	0.806	0.164	1.581
E-NSGA-II	**0.235 ± 0.055**	**0.198 ± 0.042**	1.036	0.048	2.960

Note: Bold values indicate the best result for each performance metric.

Table 3 confirms that the trajectory result was repeatable. E-NSGA-II attained the highest mean HV and lowest mean IGD. For each of the six baselines, paired two-sided Wilcoxon tests on HV and IGD gave a raw p-value of 0.00195 and a Holm-adjusted value of 0.0234, with the difference in the favorable direction for E-NSGA-II.

The same table also identified two trade-offs. First, E-NSGA-II did not obtain the lowest Spread; MODE gave the smallest value, although its substantially lower HV and higher IGD showed that this uniformity occurred over a less competitive approximation set. Second, E-NSGA-II required 2.960 s on average, compared with 0.499–1.581 s for the baselines. The proposed enhancements therefore improve front approximation at a measurable computational cost.



(**a**) Convergence behavior (hypervolume)

(**b**) Inverted generational distance (IGD)

(**c**) Spread metric (Δ)

(**d**) Mean nearest-neighbor distance (MNND)

Figure 9. Performance comparison of E-NSGA-II, NSGA-II, MOPSO, NSGA-III, MOEA/D, MODE, and IMODE across four convergence and diversity metrics over generations on large instance.

As shown in Figure 9a, E-NSGA-II achieved higher HV values during most generations and obtained the best final HV, indicating better Pareto-front approximation and wider solution coverage. In contrast, the six baselines showed limited later-stage improvement, reflecting reduced search effectiveness in the constrained CUDRP search space. Figure 9b shows that E-NSGA-II maintained a continuously decreasing IGD curve and reached the lowest final IGD, demonstrating stronger convergence toward the reference Pareto front.

These improvements resulted from the combined effects of hybrid initialization, adaptive crossover and mutation, local search, repair mechanisms, and diversity preservation, which enhanced the initial population quality, balance exploration and exploitation, improved feasibility, and maintained diversity in later generations.

Overall, the HV and IGD results demonstrate the convergence advantage of E-NSGA-II. Taken together, these results show that E-NSGA-II obtains higher-quality non-dominated solutions than the compared algorithms under the tested complex operational constraints.

4.5. Fixed-Budget Scalability and Target Attainment

To isolate the effect of problem scale from the effect of an increasing computational budget, all seven methods were tested with the same population of 60 and the same limit of 100 generations on small-, medium-, and large-scale instances. Table 4 reports only the independent validation results at the common rHV $\geq$ 0.30 threshold.

The rHV target-attainment rate was the proportion of instances whose final Pareto front achieved rHV $\geq$ 0.30. Scaling each algorithm's HV by the HV of the corresponding empirical reference front made the target comparable across instances with different objective ranges and dominated-space volumes. The indicator therefore measured whether a method reached a prescribed relative Pareto-approximation quality within the fixed budget.

Table 4. Fixed-budget computational performance and rHV target attainment.

Scale	Algorithm	Avg. Runtime (s)	Feasibility Rate
Small	E-NSGA-II	4.038	**100.0**
	NSGA-II	3.002	80.0
	MOPSO	**0.810**	30.0
	NSGA-III	3.393	80.0
	MOEA/D	2.357	70.0
	MODE	2.110	70.0
	IMODE	3.526	80.0
Medium	E-NSGA-II	5.324	**100.0**
	NSGA-II	3.261	70.0
	MOPSO	**1.011**	60.0
	NSGA-III	3.379	60.0
	MOEA/D	2.649	**100.0**
	MODE	2.308	40.0
	IMODE	3.616	**100.0**
Large	E-NSGA-II	8.431	**100.0**
	NSGA-II	4.044	50.0
	MOPSO	**1.701**	10.0
	NSGA-III	4.285	60.0
	MOEA/D	3.686	70.0
	MODE	2.790	10.0
	IMODE	4.166	30.0

Note: Bold values indicate the best result for each performance metric.

At rHV $\geq$ 0.30, the cross-algorithm median target-attainment rate decreased from 80% for small instances to 70% for medium instances and 50% for large instances, giving an overall small–medium–large deterioration under the fixed budget. E-NSGA-II was the only method that maintained 100% attainment at all three scales. At the large scale, the six baselines attained 50%, 10%, 60%, 70%, 10%, and 30% for NSGA-II, MOPSO, NSGA-III, MOEA/D, MODE, and IMODE, respectively. The strongest reductions occurred for MOPSO and the two differential-evolution baselines.

The arithmetic mean target-attainment rates were 72.9%, 75.7%, and 47.1% for small, medium, and large instances. The slight medium-over-small reversal in the mean was caused by three method-specific exceptions: MOPSO improved from 30% to 60%, while MOEA/D and IMODE improved from 70% and 80% to 100%. For MOEA/D, the medium instances provided enough trade-off structure for decomposition-based neighborhood replacement without yet overwhelming the fixed search budget; the smaller problems could under-use the fixed set of weight directions, whereas the large combinatorial space required more search. For IMODE, *p*-best guidance and success-history adaptation benefited from the richer permutation differences available at medium scale, while their effectiveness declined on the larger search space. MOPSO's non-monotonic medium result is attributable to stochastic archive-guided mutation and the sampled medium instances rather than a general scale advantage. These exceptions raised the medium mean but did not change the decreasing cross-method median or the pronounced degradation on large instances.

E-NSGA-II had the longest mean runtime at every scale, reaching 8.431 s on the large instances, compared with 1.701–4.285 s for the baselines. This overhead was consistent with its local refinement, repair, archive update, and diversity operations and should be weighed against its more stable target attainment.

4.6. Ablation Study on the Enhancement Modules

To investigate the contribution of each enhancement component in E-NSGA-II, an ablation study was conducted on a representative medium-scale instance using an NSGA-II-plus-module experimental framework. Under this framework, each module was added to the standard NSGA-II setting to examine its individual contribution, while the complete E-NSGA-II configuration combined all four modules. The experiments were performed with a population size of 100 and 200 generations. Four enhancement components were considered: hybrid initialization (*a*), adaptive genetic operators (*b*), local search (*c*), and repair/diversity preservation mechanisms (*d*). For this experiment, IGD was computed against the best-known reference set generated by E-NSGA-II, and the standard deviation-based NNS was replaced by the mean nearest-neighbor distance (MNND), which is more suitable for distinguishing between compact but clustered fronts and genuinely expanded Pareto sets. The corresponding results are summarized in Table 5.

Table 5. Final performance of the NSGA-II-based ablation settings.

Stage	HV	IGD	Spread	MNND	Runtime (s)
NSGA-II	0.297	0.163	0.902	0.052	5.623
NSGA-II+*a*	0.513	0.175	0.798	0.071	**5.471**
NSGA-II+*b*	0.335	0.191	0.874	0.067	5.678
NSGA-II+*c*	0.451	0.176	0.826	0.084	6.697
NSGA-II+*d*	0.311	0.167	**0.737**	0.070	27.754
E-NSGA-II	**0.601**	**0.083**	0.762	**0.092**	24.879

Note: Bold values indicate the best result for each performance metric.

As shown in Table 5, E-NSGA-II achieved the best overall performance among all the tested configurations in terms of solution quality, obtaining the highest HV value (0.601), the lowest IGD value (0.083), and the highest MNND value (0.092). However, this performance improvement was accompanied by a relatively high runtime of 24.879 s, which was close to the highest runtime among the tested settings. In comparison, NSGA-II+*d* achieved the best Spread value (0.737), but it also showed a relatively low HV value (0.311) and the highest runtime (27.754 s), indicating that the repair/diversity module improved distribution uniformity at a substantial computational cost when used without the other enhancement components.

Among the individual enhancement components, the hybrid initialization strategy provided noticeable improvement in HV compared with standard NSGA-II, indicating that high-quality and structurally diverse initial solutions contributed positively to the early-stage search process. The adaptive genetic operators further improved convergence behavior by dynamically adjusting crossover and mutation probabilities during evolution. Compared with standard NSGA-II, the adaptive operators contributed to improved convergence behavior during the evolutionary procedure.

The local search mechanism also contributed to solution improvement by refining route structures through neighborhood-based optimization procedures. This mechanism helped reduced inefficient routing patterns that were difficult to eliminate through standard evolutionary operators alone. In addition, the repair and diversity-preservation mechanisms improved both feasibility maintenance and population diversity. The repair mechanism helped correct infeasible solutions generated during evolution, while the diversity-preservation strategy alleviated excessive population clustering and maintained a broader distribution of non-dominated solutions.

The ablation results indicate that the enhancement modules do not operate independently. Instead, they interact complementarily throughout the evolutionary process, jointly contributing to population diversity, feasibility maintenance, and convergence improvement. Although the integration of all the enhancement modules increased computational

runtime, the overall improvement in solution quality remained evident. The ablation results therefore suggest that the proposed enhancement strategies complement each other effectively and jointly contribute to the performance improvement of E-NSGA-II.

5. Conclusions

This paper investigated a collaborative UAV–rider routing problem for urban meal delivery, jointly coordinating restaurant pickups, UAV transport, rendezvous assignments, and rider last-mile routes. The model minimizes operational cost, makespan, emissions, and average tardiness under soft customer time windows. Its main methodological contribution is an E-NSGA-II integrating hybrid initialization, adaptive variation, problem-specific local search, deterministic repair, and archive-assisted diversity preservation.

Our experiments showed that E-NSGA-II achieved the strongest overall HV and IGD performance against six representative multi-objective algorithms. Under fixed-budget scalability tests, it attained rHV ≥ 0.30 on all small, medium, and large instances, while baseline performance deteriorated more substantially as problem size increased. The ablation results indicate that the individual modules had complementary and metric-dependent effects: the full configuration yielded the best IGD and HV.

The study remained limited by static orders, deterministic preparation and travel times, synthetic instances, simplified rendezvous synchronization, and the absence of detailed wind, airspace, battery, and energy-consumption models. Future work should therefore consider dynamic and stochastic rolling-horizon optimization with real operational data, explicit hand-off timing, order-level flow tracking, and more realistic UAV energy and airspace constraints. Integrating delay prediction [42], endogenous demand responses [43], parallel evaluation, surrogate models, and adaptive computational budgets may further improve planning accuracy and operational efficiency.

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

Funding: This work was supported by the Natural Science Foundation of Sichuan Province (No. 2025NSFSC2001), the Hangzhou Key Scientific Research Program (No. 2025SZDA0325) and the Open Project Program of State Key Laboratory of CNS/ATM (No. 2025B32).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data generated and analyzed during the current study are available from the corresponding author upon reasonable request.

Acknowledgments: The authors would like to thank the reviewers and editors for their constructive comments.

Conflicts of Interest: Author Yiwu Feng was employed by the company Huadian Electric Power Research Institute Co., Ltd. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

Variable	Definition
x_{ijk}^{U}	Binary variable equal to 1 if UAV k travels from node i to node j, and 0 otherwise.
x_{ijk}^{R}	Binary variable equal to 1 if rider k travels from node i to node j, and 0 otherwise.

y_{rt}^{U}	Binary variable equal to 1 if the package collected from restaurant r is transferred at rendezvous point t.
z_{ct}^{R}	Binary variable equal to 1 if customer c is served through rendezvous point t.
w_{ctk}^{R}	Binary variable equal to 1 if customer c is served by rider k from rendezvous point t.
δ_{k}^{U}	Binary UAV-activation variable.
δ_{k}^{R}	Binary rider-activation variable.
δ_{tk}^{R}	Binary variable equal to 1 if rider k is assigned to rendezvous point t.
a_{ik}^{U}, a_{ik}^{R}	Arrival times of UAV/rider k at node i.
d_{ik}^{U}, d_{ik}^{R}	Departure times of UAV/rider k from node i.
L_{ik}^{U}, L_{ik}^{R}	Loads of UAV/rider k after leaving node i.
ξ_{c}	Non-negative tardiness slack for customer c.
C_{k}^{U}, C_{k}^{R}	Completion times of UAV/rider k.
$T_{\max}$	Overall maximum completion time of the collaborative operation.
Notation	**Description**
V_D	UAV depot (single location), $V_D = \{0\}$;
V_R	Restaurant nodes (pickup locations), $V_R = \{1, 2, \ldots, M\}$;
V_C	Customer nodes (delivery locations), $V_C = \{M+1, M+2, \ldots, M+N\}$;
V_T	Rendezvous points (transfer locations), $V_T = \{M+N+1, \ldots, M+N+P\}$;
V	$V = V_D \cup V_R \cup V_C \cup V_T$ (All vertices);
K^U	Set of UAVs, $K^U = \{1, 2, \ldots, K_U\}$;
K^R	Set of riders, $K^R = \{1, 2, \ldots, K_R\}$;
i, j	Indices for nodes;
k	Index for vehicles (UAVs or riders);
r	Index for restaurants;
c	Index for customers;
t	Index for rendezvous points.
d_i	Demand at node i ($d_0 = 0$, $d_t = 0\ \forall t \in V_T$);
q_r^p	Preparation time at restaurant r;
$[e_c, l_c]$	Time window for customer c;
s_i	Service time at node i;
Q_U	UAV payload capacity (homogeneous);
Q_R	Rider carrying capacity (homogeneous);
E_U	UAV endurance time limit (min);
v_U	UAV average speed;
v_R	Rider average speed;
$dist_{ij}^{U}$	Distance between nodes i and j for UAVs;
$dist_{ij}^{R}$	Distance between nodes i and j for riders;
t_{ij}^{U}	$t_{ij}^{U} = dist_{ij}^{U}/v_U$, travel time for UAV;
t_{ij}^{R}	$t_{ij}^{R} = dist_{ij}^{R}/v_R$, travel time for rider;
c_{ij}^{U}	Operational cost for UAV on arc (i, j);
c_{ij}^{R}	Operational cost for rider on arc (i, j);
FC_U	Fixed cost per UAV used;
FC_R	Fixed cost per rider used;
em_{ij}^{U}	Carbon emissions for UAV on arc (i, j);
em_{ij}^{R}	Carbon emissions for rider on arc (i, j);
$\mathcal{M}$	A sufficiently large positive constant used in conditional constraints.
x_{ijk}^{U}	Binary, 1 if UAV k travels from i to j, 0 otherwise;
x_{ijk}^{R}	Binary, 1 if rider k travels from node i to node j, 0 otherwise;
y_{rt}^{U}	Binary, 1 if package from restaurant r is transferred at rendezvous point t, 0 otherwise;
z_{ct}^{R}	Binary, 1 if customer c is served via rendezvous point t, 0 otherwise;
w_{ctk}^{R}	Binary, 1 if customer c is served by rider k from rendezvous point t, 0 otherwise;
δ_{k}^{U}	Binary, 1 if UAV k is activated, 0 otherwise;
δ_{k}^{R}	Binary, 1 if rider k is activated, 0 otherwise;

δ_{tk}^{R}	Binary, 1 if rider k is assigned to rendezvous point t, 0 otherwise;
a_{ik}^{U}, a_{ik}^{R}	Arrival time of UAV/rider k at node i;
d_{ik}^{U}, d_{ik}^{R}	Departure time of UAV/rider k from node i;
ξ_c	Non-negative tardiness slack of customer c;
L_{ik}^{U}	Load of UAV k after leaving node i;
L_{ik}^{R}	Load of rider k after leaving node i;
C_k^{U}, C_k^{R}	Completion time of UAV/rider k;
$T_{\max}$	Overall maximum completion time of the collaborative delivery operation.

References

1. Vaigandla, K.K.; Thatipamula, S.; Karne, R.K. Investigation on unmanned aerial vehicle (UAV): An overview. *IRO J. Sustain. Wirel. Syst.* **2022**, *4*, 130–148. [CrossRef]
2. Sharma, R.; Arya, R. UAV based long range environment monitoring system with Industry 5.0 perspectives for smart city infrastructure. *Comput. Ind. Eng.* **2022**, *168*, 108066. [CrossRef]
3. Mohsan, S.A.H.; Othman, N.Q.H.; Li, Y.; Alsharif, M.H.; Khan, M.A. Unmanned aerial vehicles (UAVs): Practical aspects, applications, open challenges, security issues, and future trends. *Intell. Serv. Robot.* **2023**, *16*, 109–137. [CrossRef] [PubMed]
4. Nguyen-Phuoc, D.Q.; Nguyen, N.A.N.; Nguyen, M.H.; Nguyen, L.N.T.; Oviedo-Trespalacios, O. Factors Influencing Road Safety Compliance among Food Delivery Riders: An Extension of the Job Demands–Resources (JD-R) Model. *Transp. Res. Part A Policy Pract.* **2022**, *166*, 541–556. [CrossRef]
5. Huang, C.; Fang, S.; Wu, H.; Wang, Y.; Yang, Y. Low-altitude intelligent transportation: System architecture, infrastructure, and key technologies. *J. Ind. Inf. Integr.* **2024**, *42*, 100694. [CrossRef]
6. Calamoneri, T.; Corò, F.; Mancini, S. Management of a post-disaster emergency scenario through unmanned aerial vehicles: Multi-depot multi-trip vehicle routing with total completion time minimization. *Expert Syst. Appl.* **2024**, *251*, 123766. [CrossRef]
7. Jung, S. MILP-based cost and time-competitive vehicle routing problem for last-mile delivery service using a swarm of UAVs and UGVs. *J. Air Transp. Manag.* **2025**, *124*, 102736. [CrossRef]
8. Jasim, A.N.; Fourati, L.C. Guided genetic algorithm for solving capacitated vehicle routing problem with unmanned-aerial-vehicles. *IEEE Access* **2024**, *12*, 106333–106358. [CrossRef]
9. Ma, M.; Wang, C.; Li, Z.; Liu, F. A Proactive Resource Allocation Algorithm for UAV-Assisted V2X Communication Based on Dynamic Multi-Objective Optimization. *IEEE Commun. Lett.* **2024**, *28*, 2814–2818. [CrossRef]
10. Zhang, C.; Sun, G.; Li, J.; Wu, Q.; Wang, J.; Niyato, D.; Liu, Y. Multi-Objective Aerial Collaborative Secure Communication Optimization via Generative Diffusion Model-Enabled Deep Reinforcement Learning. *Mob. Comput. IEEE Trans.* **2025**, *24*, 3041–3058. [CrossRef]
11. Akopov, A.S.; Beklaryan, L.A.; Thakur, M. Improvement of Maneuverability Within a Multiagent Fuzzy Transportation System With the Use of Parallel Biobjective Real-Coded Genetic Algorithm. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 12648–12664. [CrossRef]
12. Zhang, Q.; Li, H. MOEA/D: A Multiobjective Evolutionary Algorithm Based on Decomposition. *IEEE Trans. Evol. Comput.* **2007**, *11*, 712–731. [CrossRef]
13. Deb, K.; Jain, H. An Evolutionary Many-Objective Optimization Algorithm Using Reference-Point-Based Nondominated Sorting Approach, Part I: Solving Problems With Box Constraints. *IEEE Trans. Evol. Comput.* **2014**, *18*, 577–601. [CrossRef]
14. Xue, F.; Sanderson, A.C.; Graves, R.J. Pareto-Based Multi-Objective Differential Evolution. In *Proceedings of the 2003 Congress on Evolutionary Computation*; IEEE: New York, NY, USA, 2003; Volume 2, pp. 862–869.
15. Sallam, K.M.; Elsayed, S.M.; Chakrabortty, R.K.; Ryan, M.J. Improved Multi-Operator Differential Evolution Algorithm for Solving Unconstrained Problems. In *Proceedings of the 2020 IEEE Congress on Evolutionary Computation*; IEEE: New York, NY, USA, 2020.
16. He, M.; Wang, Z.; Chen, H.; Cao, Y.; Ma, L. Multi-Objective Evolutionary Algorithm Based on Decomposition With Orthogonal Experimental Design. *Expert Syst.* **2025**, *42*, e13802.
17. Caballero, A.; Roman-Escorza, F.J.; Maza, I.; Ollero, A. A Multi-UAV Approach for Fast Inspection of Overhead Power Lines: From Route Planning to Field Operation. *J. Intell. Robot. Syst.* **2025**, *111*, 67. [CrossRef]
18. Divkoti, M.R.R.; Nouri-Baygi, M. RCS: A fast path planning algorithm for unmanned aerial vehicles: RCS: A fast path planning algorithm for unmanned aerial vehicles. *Soft Comput.* **2025**, *29*, 275–298. [CrossRef]
19. Yang, G.; Mo, Y.; Lv, C.; Zhang, Y.; Li, J.; Wei, S. A dual-layer task planning algorithm based on UAVs-human cooperation for search and rescue. *Appl. Soft Comput.* **2025**, *181*, 113488. [CrossRef]
20. Gonzalez-R, P.L.; Canca, D.; Andrade-Pineda, J.L.; Calle, M.; Leon-Blanco, J.M. Truck-drone team logistics: A heuristic approach to multi-drop route planning. *Transp. Res. Part C Emerg. Technol.* **2020**, *114*, 657–680. [CrossRef]

21. Long, Y.; Xu, G.; Zhao, J.; Xie, B.; Fang, M. Dynamic truck–UAV collaboration and integrated route planning for resilient urban emergency response. *IEEE Trans. Eng. Manag.* **2023**, *71*, 9826–9838. [CrossRef]
22. Kim, S.B.; Lee, S.H. Battery Balancing Algorithm for an Agricultural Drone Using a State-Of-Charge-Based Fuzzy Controller. *Appl. Sci.* **2020**, *10*, 5277. [CrossRef]
23. Liao, Z.; Li, S.; Zhou, P.; Zhang, C. Decay regularized stochastic configuration networks with multi-level data processing for UAV battery RUL prediction. *Inf. Sci.* **2025**, *701*, 121840. [CrossRef]
24. Joo, S.; Kim, H.; Son, H. Multiobjective design optimization of hybrid-powered multirotors for enhanced endurance and payload capacity. *IEEE/ASME Trans. Mechatronics* **2024**, *30*, 1959–1970. [CrossRef]
25. Cheng, N.; Wu, S.; Wang, X.; Yin, Z.; Li, C.; Chen, W.; Chen, F. AI for UAV-assisted IoT applications: A comprehensive review. *IEEE Internet Things J.* **2023**, *10*, 14438–14461. [CrossRef]
26. Sekander, S.; Tabassum, H.; Hossain, E. Multi-Tier Drone Architecture for 5G/B5G Cellular Networks: Challenges, Trends, and Prospects. *IEEE Commun. Mag.* **2018**, *56*, 96–103. [CrossRef]
27. Chen, C.; Cheng, Y.; Dou, Z.; Li, H.; Liu, L.; Pei, Q.; Wan, S. An UAV Power Line Patrolling System With Edge Intelligence and Beidou SMS in Signal Loss Area. *IEEE Internet Things J.* **2025**, *12*, 30020–30035. [CrossRef]
28. González-Jorge, H.; Martínez-Sánchez, J.; Bueno, M.; Arias, P. Unmanned Aerial Systems for Civil Applications: A Review. *Drones* **2017**, *1*, 2. [CrossRef]
29. Outay, F.; Mengash, H.A.; Adnan, M. Applications of unmanned aerial vehicle (UAV) in road safety, traffic and highway infrastructure management: Recent advances and challenges. *Transp. Res. Part A Policy Pract.* **2020**, *141*, 116–129. [CrossRef] [PubMed]
30. Zhou, Y.; Rui, T.; Li, Y.; Zuo, X. A UAV patrol system using panoramic stitching and object detection. *Comput. Electr. Eng.* **2019**, *80*, 106473. [CrossRef]
31. Wang, Z.; Chen, T.; Chen, J. Human-UAV interactive perception: Skeleton-based iterative perspective optimization algorithm for UAV patrol tracking of large-scale pedestrian abnormal behavior. *Appl. Soft Comput.* **2024**, *167*, 112467. [CrossRef]
32. Qadir, Z.; Ullah, F.; Munawar, H.S.; Al-Turjman, F. Addressing disasters in smart cities through UAVs path planning and 5G communications: A systematic review. *Comput. Commun.* **2021**, *168*, 114–135. [CrossRef]
33. Yucesoy, E.; Balcik, B.; Coban, E. The role of drones in disaster response: A literature review of operations research applications. *Int. Trans. Oper. Res.* **2025**, *32*, 545–589.
34. Palmeri, V.; Stefano, C.D.; Guida, G.; Nicosia, A.; Ferro, V. Monitoring the temporal evolution of a Sicilian badland area by unmanned aerial vehicles. *Geomorphology* **2024**, *466*, 109443. [CrossRef]
35. Qiao, B.; Wang, L.; Han, H.; Huang, Z.; Lin, Y.; Jia, Z.; Nie, H.; He, X.; Liu, Y. Effects of airflow disturbance from a multi-rotor unmanned aerial vehicle on fruit tree frost protection. *Biosyst. Eng.* **2024**, *241*, 95–112. [CrossRef]
36. Ong, J.W.; Abid, H.A.; Minifie, T.; Lin, E.S.; Song, Z.; Katariya, M.; Liew, O.; Ng, T. Unmanned aerial vehicle transport of frozen blood samples using phase change materials. *Biosyst. Eng.* **2022**, *221*, 30–42. [CrossRef]
37. Liu, Y. An optimization-driven dynamic vehicle routing algorithm for on-demand meal delivery using drones. *Comput. Oper. Res.* **2019**, *111*, 1–20. [CrossRef]
38. Pinto, R.; Zambetti, M.; Lagorio, A.; Pirola, F. A network design model for a meal delivery service using drones. *Int. J. Logist. Res. Appl.* **2020**, *23*, 354–374.
39. Hamid, M.; Nasiri, M.M.; Rabbani, M. A mixed closed-open multi-depot routing and scheduling problem for homemade meal delivery incorporating drone and crowd-sourced fleet: A self-adaptive hyper-heuristic approach. *Eng. Appl. Artif. Intell.* **2023**, *120*, 105876. [CrossRef]
40. Hana, H.; Kob, Y.D. Optimal drone delivery scheduling with split deliveries for multiple restaurants using a shared fleet. *Comput. Ind. Eng.* **2025**, *212*, 111663. [CrossRef]
41. Lu, F.; Jiang, R.; Bi, H.; Gao, Z. Order Distribution and Routing Optimization for Takeout Delivery under Drone–Rider Joint Delivery Mode. *J. Theor. Appl. Electron. Commer. Res.* **2024**, *19*, 774–796. [CrossRef]
42. Sun, B.; Xu, Z.; Ranjitkar, P.; Wei, M. DS-MGCSTNet: Multi-Graph Convolution Spatial-Temporal Network with Dynamic and Static Graph Fusion for Flight Delay Prediction. *J. Air Transp. Manag.* **2026**, *136*, 103047. [CrossRef]
43. Wiseman, Y. Autonomous Vehicles Will Spur Moving Budget from Railroads to Roads. *Int. J. Intell. Unmanned Syst.* **2024**, *12*, 19–31.

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.