Journal of Industrial and Management Optimization

JIMO, 22(8): 3789–3834.
DOI: 10.3934/jimo.2026136
Received: 13 April 2026
Revised: 27 May 2026
Accepted: 02 July 2026
Published: 21 July 2026

https://www.aimspress.com/journal/jimo

Research article

Collaborative route optimization for urban emergency vehicles and drones under dynamic evolution of heavy rainfall-induced urban flooding

Yuanbo Zhang and Na Li*

School of Economics and Management, Tianjin Chengjian University, Tianjin, China

* **Correspondence:** Email: llnn49@tcu.edu.cn.

Abstract: Urban flooding increasingly impairs road network capacity and delays ground-based emergency rescue operations. To address the fundamental conflict between dynamic road obstructions and rescue urgency, we proposed a novel coordinated rescue framework integrating emergency vehicles and drones. A multi-objective mixed-integer programming model was formulated to minimize total rescue time, total cost, and average time-window penalty. The model incorporated a vehicle speed function governed by real-time inundation depth and a time-varying drone ground speed model based on wind speed and direction, collectively capturing the dynamic operational environment. Meteorological risks were further quantified within the collaborative path optimization module, with drone flight safety guaranteed through cumulative risk thresholds. To solve this NP-hard problem, an enhanced adaptive large neighborhood search (EALNS) algorithm was developed, integrating a predictive search mechanism, problem-oriented operator design, and local search enhancement strategies to improve solution efficiency and quality in complex dynamic environments. The validity of the proposed model and algorithm was assessed through numerical experiments based on multiple benchmark instances and a real-world case study from Tianjin, China. Our experimental results demonstrated that EALNS achieved average improvements of 45%, 10.9%, and 37.5% over the genetic algorithm, the standard adaptive large neighborhood search algorithm, and the whale optimization algorithm, respectively. Furthermore, compared with the vehicle-only mode, the collaborative rescue mode reduced total rescue time, total cost, and average time-window penalty by 11.2%, 10.0%, and 39.4%, respectively. These results collectively validate the effectiveness and practical applicability of the proposed approach under dynamic disaster conditions.

Keywords: vehicle-drone collaborative delivery; urban flooding; Emergency response; dynamic routing; ALNS
Mathematics Subject Classification: 90B06, 90C29

1. Introduction

In recent years, against the backdrop of global warming, the frequency of extreme rainfall events has increased significantly, with flooding emerging as one of the most severe natural disasters worldwide [1]. The impacts of such disasters are particularly concentrated in urban areas [2]. Notable examples include the Beijing "21 July" torrential rain event in 2012 [3], the catastrophic flooding in Derna, Libya, on 10 September 2023 [4], and the Dubai flooding on 15 April 2024 [5]. These extreme events not only caused substantial economic losses and casualties but also posed severe challenges to emergency response operations. Urban road networks, characterized by their openness and complexity, are particularly susceptible to disruption from external factors such as extreme weather. During prolonged rainfall, these networks undergo a progressive transition from localized to widespread failures, significantly impeding emergency response efficiency [6–8]. Consequently, achieving efficient emergency response in dynamic urban flooding environments has become a critical issue requiring urgent attention.

Traditional ground-based rescue operations face significant challenges during extreme weather events. Road flooding reduces accessibility, preventing rescue vehicles from reaching core disaster zones [9,10], while rescue personnel face heightened safety risks. Drone technology, characterized by high maneuverability, independence from ground traffic constraints, and rapid deployment capabilities, offers promising solutions to these operational bottlenecks Faiz et al. [11]. Furthermore, the coordinated deployment of emergency vehicles and drones can effectively exploit the complementary strengths of both modes of transport: Emergency vehicles facilitate the bulk transport of relief supplies, whereas drones enable precise deliveries to waterlogged or isolated areas inaccessible to ground vehicles. This integrated approach enhances overall system coverage, response efficiency, and operational resilience.

Against this backdrop, we develop a mixed-integer programming (MIP) model to optimize the coordinated routing of emergency vehicles and drones based on the dynamic evolution of urban flooding during heavy rainfall. To enhance computational efficiency, an Enhanced Adaptive Large Neighborhood Search (EALNS) algorithm is proposed. The model is validated through numerical experiments using Zhongbei Town, Xiqing District, Tianjin as a case study to examine the optimization of coordinated vehicle-drone routing under dynamically evolving flood conditions. The major contributions of this study are as follows:

(1) In terms of modeling, we develop a multi-objective optimization model that minimizes the total rescue time, total cost, and average penalty. The model incorporates a dynamic vehicle speed function influenced by water depth and a soft time window penalty mechanism, designed to capture the time-varying characteristics of road passability as flooding evolves, thereby enhancing the realism of emergency rescue simulations.

(2) In terms of algorithm design, to address the computational complexity inherent in this NP-hard problem, the EALNS algorithm is proposed. This algorithm integrates dynamic operator selection,

problem-specific operator design, and local search enhancement strategies to achieve efficient solutions for large-scale instances.

(3) We account for the effects of rainstorm-induced urban waterlogging on drone performance by developing a time-varying drone ground speed model based on wind velocity and direction. The model captures the dynamic influence of meteorological conditions on drone flight time and operational risk, thereby enhancing the applicability and accuracy of the collaborative scheduling model in real-world dynamic environments.

The remainder of this paper is organized as follows: In Section 2, we provide a systematic review of the relevant literature. In Section 3, we formally define the problem and establish the mathematical model. In Section 4, we present the EALNS algorithm framework and its key mechanisms. In Section 5, we validate the proposed model and algorithm through computational experiments and a case study. In Section 6, we conclude the study and outline directions for future research.

2. Literature review

There has been extensive research on emergency relief supply delivery [12–14]. To systematically review progress in this field, in this section, we discuss developments in four distinct categories: drone delivery in emergency response, vehicle–drone collaborative delivery systems, dynamic emergency response operations, and route optimization algorithms.

2.1. Drone delivery in emergency response

In post-disaster emergency response, conventional transport methods, particularly ground vehicles, frequently experience partial or complete disruption due to severely damaged road networks or waterlogged conditions. In such scenarios, drones offer high maneuverability, rapid response capabilities, and the ability to operate independently of ground traffic constraints, enabling them to execute missions in complex, hazardous, or inaccessible environments [15]. Drones have demonstrated significant value in disaster relief operations [16] and are increasingly deployed across stages, including disaster reconnaissance and the precise delivery of blood and other critical rescue supplies [17]. Macrina et al. [18] categorized and analyzed studies on the last-mile delivery problem based on different drone delivery models, exploring corresponding solutions for each. Du et al. [19] addressed the route planning problem for medical supply delivery drones during major public health emergencies by developing a mixed-integer programming model to mitigate risks and enhance efficiency. Ghelichi et al. [20] proposed an optimization model for drone fleet logistics, enabling the timely delivery of medical supplies to remote areas. Studies have demonstrated the potential of drones to improve accessibility and response efficiency in emergency logistics. However, most researchers treat drones as independent delivery resources and primarily focus on static route planning or supply allocation. In contrast, we integrate drones into a coordinated vehicle–drone rescue framework for urban flooding scenarios. Moreover, we account for the dynamic effects of road inundation, wind speed, wind direction, and rainfall-induced flight risks, thereby more accurately capturing the operational complexity of emergency rescue under extreme weather conditions.

2.2. Vehicle–drone collaborative delivery systems

Given the limitations of drones in aspects such as mission duration constrained by battery life [21], researchers have begun exploring scheduling schemes that integrate drones with conventional ground vehicles [22]. Within this coordinated system, trucks and drones complement each other's strengths. Trucks serve as mobile supply stations for drones and handle bulk cargo transportation, while drones depart from trucks or distribution centers to deliver goods rapidly, executing precise delivery tasks to isolated or hard-to-reach areas. This integration enhances the coverage breadth and response agility of rescue logistics networks. Wu et al. [23] proposed an improved Variable Neighborhood Descent (IVND) method to address collaborative route planning for trucks and drones in pandemic contexts. Huang et al. [24] introduced an ant colony optimization algorithm to optimize customer allocation and dispatch frequency for drone and truck fleets. To address drone endurance limitations, Ribeiro et al. [25] developed a mixed-integer linear programming model incorporating synchronous utilization of drones and mobile charging stations and proposed a heuristic method for constructing and adjusting flight routes to deliver efficient, rapid responses during search-and-rescue operations. Considering the dynamic nature and uncertainty of resource transportation, Peng et al. [26] proposed a multi-agent deep reinforcement learning-based collaborative real-time route planning method for truck–drone coordination. By modeling the problem as a Markov game, they effectively optimized transport costs, delivery timeliness, and service fairness.

Various models have been developed to enhance delivery systems. Murray and Raj [27] introduced a method for coordinating a single truck with a heterogeneous drone fleet to deliver small parcels to geographically dispersed customers, where each drone can launch from the truck to serve a single customer before returning to rendezvous with the truck for reloading or transport to a new launch site. Salama and Srinivas [28] proposed a novel truck–drone coordination paradigm permitting trucks to dock at non-customer locations for drone launch and recovery operations. Their model also addresses vehicle assignment to customer locations, route planning for trucks and drones, and the scheduling of drone launch/recovery activities alongside truck operator tasks at each docking point. Thomas et al. [29] relaxed traditional constraints by permitting drones to autonomously launch and land on moving trucks within the drone scheduling and en route operations problem for single-truck multi-drone delivery systems. Zhou et al. [30] introduced a novel system where multiple vehicles and drones collaborate to serve customers, enabling drones to make multiple round trips when paired vehicles are stationary at customer nodes. Dukkanci et al. [31] proposed an emergency supply delivery scheme integrating trucks with large-capacity and small-capacity drones. Within this system, large-capacity drones transport supplies directly from warehouses to target disaster sites, while trucks loaded with small drones and supplies travel along predefined routes to regional transfer stations, from which small drones complete the final leg of delivery to end-user locations, effectively addressing transport bottlenecks where road conditions prevent direct truck access. In collaborative delivery systems, the diversity of resource types and performance evaluation in dynamic environments are equally critical. Gong et al. [32] developed a coordinated optimization model for emergency supply distribution in flood-affected areas, assigning trucks to land demand points, speedboats to water demand points, and drones to remote or hard-to-access points. Their improved adaptive large neighborhood search algorithm incorporates refined deletion and insertion operators, which reduces total delivery time and enhances multi-modal collaboration efficiency, demonstrating the effectiveness of coordinated multi-modal logistics under flood conditions. Arumugaraj Nadar et al. [33] proposed a hypercube queuing model together with approximation methods for emergency medical services, explicitly incorporating the matching constraints between multi-type vehicles and patients. In home healthcare delivery, service compatibility constraints combined with time window requirements give rise to complex vehicle routing optimization problems. Acar and Altin [34] formulated the problem as the multi-depot generalized colored traveling salesman problem with time windows and solved it using a variable neighborhood descent (VND) algorithm. Recognizing that drones can serve multiple customers per flight, Luo et al. [35] developed a model demonstrating that drones can perform launch and recovery operations at warehouses and any customer node. Gu et al. [36] proposed a multi-visit vehicle routing problem where each vehicle carries a drone capable of servicing multiple clients per flight. Due to the heavy and urgent nature of the relief supplies and considering the safety of the truck drivers, we propose a new delivery system that is different from the other studies. Research in vehicle–drone collaborative delivery has largely focused on parcel transportation, medical logistics, and general emergency supply distribution. While these studies offer valuable insights for truck–drone coordination, they often assume stable road conditions and fixed drone flight parameters. In comparison, in this study, we consider urban flood rescue, where vehicle speeds may fluctuate with real-time inundation depths, and drone operations are affected by wind speed, wind direction, and rainfall-related flight risks. As a result, the proposed framework provides a more realistic representation of the dynamic and hazardous conditions encountered in post-flood emergency rescue operations.

2.3. Dynamic emergency response operations

Post-disaster rescue environments exhibit pronounced dynamic and uncertain characteristics, including real-time fluctuations in demand, continuous evolution of transport networks, and variable resource availability. Consequently, dynamic optimization methods have gained increasing attention in the field of emergency supply distribution. Zhong et al. [37] developed a risk-aversion model integrating facility location and vehicle routing, employing conditional value at risk (CVaR) to quantify dynamic risks in decision-making. This approach provides valuable reference for formulating relief strategies that are efficient and robust under conditions of incomplete information. Cao et al. [38] developed a multi-period fuzzy bi-level optimization model addressing the dynamic evolution of post-disaster rescue environments. This model synergistically optimizes resource allocation while balancing urgent demands and sustainability objectives within humanitarian supply chains. Zhao et al. [39] formulated a two-stage decision model with the objective of minimizing transport time and road congestion. The model employs historical data to predict travel times, thereby determining dynamic routes for emergency vehicles. Ehmke et al. [40] examined the dynamic impact of time-varying traffic speeds and vehicle loads on route emissions in urban logistics. By integrating large-scale real-world speed data, they analyzed the dynamic and time-varying characteristics of vehicle speeds on urban roads and proposed a tabu search algorithm for solving the problem. Liu et al. [41] addressed post-disaster relief supply distribution under demand uncertainty and time-evolving transport networks using a multi-period stochastic programming approach. Avishan et al. [42] proposed an adjustable robust optimization method to generate routes and determine service times for rescue logistics teams under travel time uncertainty. Their model permits rescue teams to adjust service decisions based on information revealed during operations. Yang et al. [43] constructed a two-stage stochastic programming model for post-disaster rescue scenarios, incorporating a dynamic drone allocation scheme to address uneven material demand. Building on these advances, researchers have further integrated real-time traffic data and meteorological warnings. K. Li et al. [44] proposed improved bidirectional ant colony optimization and discrete honey badger algorithms to address multi-drone path planning under dynamic environments with multiple wind field conditions. Chen et al. [45] incorporated stochastic time-dependent travel times to capture the time-varying characteristics of traffic conditions and empirically validated their model on the Vienna road network to optimize vehicle routes and departure times under stochastic demand and time-varying travel time uncertainty. Researchers have also increasingly explored dynamic truck–drone routing in disaster-response contexts. Li et al. [46] formulated a collaborative truck–drone vehicle routing problem with mixed line-haul and backhaul operations during hurricanes, in which wind-speed fluctuations constrain the availability of trucks and drones and adjust their travel times. To address these dynamics, they embedded a resilient adaptive large neighborhood search algorithm within a rolling-horizon framework, enabling routing decisions to be updated as new requests arrive and wind conditions change. Shi et al. [47] examined emergency-supply delivery under secondary disasters by developing a dynamic multi-trip vehicle routing problem with drones for a hybrid fleet of regular trucks and truck–drone tandems. Their model incorporates updated disaster information, including newly affected sites, demand changes, and road-risk variations, updating the remaining routes accordingly. Collectively, research on dynamic emergency response has addressed demand uncertainty, stochastic travel times, traffic congestion, dynamic resource allocation, and the challenges of dynamic truck–drone routing under disaster conditions. Relatively few researchers have examined the combined effects of flood-induced road inundation on vehicle mobility and meteorological conditions on drone flight performance. To help address these challenges, we model vehicle speed as a function of real-time inundation depth and develop a time-varying drone ground speed model based on wind speed and direction while incorporating rainfall-related flight risk constraints.

2.4. Route optimization algorithms

Route optimization problems, particularly vehicle routing problems (VRPs) and their variants, have attracted extensive research attention, prompting the development of numerous algorithms designed to address them. These algorithms can be broadly categorized into three classes: exact algorithms, heuristic algorithms, and learning-based algorithms.

Exact algorithms offer distinct advantages when solving small-scale or well-structured VRP instances. Costa et al. [48] employed a branch-and-cut approach to solve vehicle routing problems. Vásquez et al. [49] developed an improved Benders decomposition algorithm for the drone-truck collaborative route planning problem by introducing efficient inequalities and optimization cuts derived from the problem structure. In last-mile delivery, dynamically adjusting the set of delivery points to reduce problem scale has proven to be an effective strategy. Saeed Osman [50] developed a dynamic MIP decomposition method that iteratively updates the active delivery point set, thereby substantially reducing the number of variables and constraints in multi-period delivery problems. However, exact algorithms exhibit inherent limitations when addressing dynamic demands, large-scale instances, and complex constraints, prompting researchers to explore more scalable heuristic approaches. Heuristic algorithms, characterized by their robust global search capabilities and adaptability to complex problems, have become among the most widely applied methods in route optimization research. Motaghedi-Larijani [51] employed NSGA-II and multi-objective simulated annealing to optimize cross-dock opening frequencies. Ant colony optimization and genetic algorithms have also been extensively applied. Li et al. [52] introduced an enhanced ant colony algorithm incorporating a fuzzy logic memory mechanism and a hierarchical route construction method to improve the feasibility rate of solution generation, demonstrating strong applicability in real-world terrain environments. Peng et al. [53] proposed a hybrid genetic algorithm to facilitate efficient parcel delivery through the coordinated operation of ground vehicles and multiple unmanned aerial vehicles. Among metaheuristics, the adaptive large neighborhood search (ALNS) algorithm has garnered significant attention due to its flexible operator scheduling mechanism and robust local search capabilities. Özarık et al. [54] developed a tailored ALNS heuristic for last-mile delivery problems characterized by stochastic customer accessibility and potential multiple visits. Meng et al. [55] proposed a customized ALNS algorithm for time-window-constrained multiple-visit drone-assisted pick-up and delivery tasks, demonstrating robust performance through extensive numerical experiments. Cheng et al. [56] designed an improved ALNS (IALNS) algorithm for truck-drone-motorcycle coordination in mountain wildfire rescue scenarios, outperforming multiple benchmark heuristics in computational efficiency and solution quality. Recent years have witnessed significant advances in route optimization through deep learning and reinforcement learning approaches. Li et al. [57] proposed an end-to-end deep reinforcement learning framework for multi-objective optimization. Guan et al. [58] employed a multi-head collaborative attention mechanism, demonstrating excellent generalization performance across problems of varying scales and distributions. For dynamic VRPs, Akkerman et al. [59] provided systematic guidance on applying neural network-based reinforcement learning, comparing learning strategies for handling stochastic customer requests. Moreover, Zhou et al. [60] proposed a Priority-aware Distributed Proximal Policy Optimization (PD_PPO) for multi-vessel cooperative rescue in flood-disaster scenarios. By combining graph-based environment modeling, multi-attribute task-priority evaluation, and distributed reinforcement learning, the method enables vessels to prioritize high-value tasks under dynamic hydrological conditions while reducing the growth of the joint action space. This work demonstrates the potential of RL-based approaches for adaptive emergency scheduling. Nevertheless, learning-based methods continue to face challenges, including strong dependence on training data and limited capability in handling complex constraints. Building on research, exact algorithms perform well for small-scale routing problems, while heuristic and metaheuristic algorithms are generally better suited for complex, large-scale scenarios. On this basis, we develop an enhanced adaptive large neighborhood search algorithm that integrates predictive search, problem-oriented operators, and local search strategies, aiming to improve solution quality and computational efficiency in dynamic rescue operations.

In summary, researchers have conducted a relatively comprehensive exploration of optimization problems in emergency vehicle and drone rescue operations, encompassing route planning under static and dynamic scenarios, single-objective and multi-objective optimization, and the incorporation of diverse constraints and objectives in model construction and algorithm development. Although the problem addressed in this paper bears certain formal similarities to the Two-Echelon Vehicle Routing Problem (Two-Echelon VRP) and the Truck-Drone Routing Problem, the research focus differs considerably. Specifically, Two-Echelon VRP studies predominantly entail static network structures and transshipment optimization, whereas we concentrate on dynamic process control under extreme weather conditions, modeling emergency vehicle transit times as time-varying functions of inundation depth and treating drone flight times as subject to the dynamic effects of wind velocity vector composition. The central challenge lies in coordinating the collaborative scheduling of two transport networks and modes within a highly dynamic environment. Building on this foundation, we introduce a water-depth-driven vehicle speed function, a time-varying drone ground speed model driven by wind speed and direction, and flight risk threshold constraints triggered by rainfall intensity, ensuring that drone path planning balances not only distance but also timeliness and operational safety. Furthermore, a multi-objective optimization framework encompassing time, cost, and penalty terms is adopted, and predictive mechanisms are incorporated to enhance dynamic adaptability and support real-time response in disaster scenarios. Nevertheless, research on dynamic route selection largely entails the influence of traffic flow variations on vehicle speeds while giving insufficient attention to the mechanisms by which sustained dynamic disturbances, such as road flooding induced by heavy rainfall, affect passage efficiency. Additionally, researchers commonly adopt static parameters for drone flight speed, with limited consideration of the dynamic effects of meteorological factors, such as wind speed and direction, on drone ground speed. To address these gaps, we model the dynamic variation in vehicle travel speeds arising from road inundation under rainstorm conditions and construct a time-varying speed matrix. A time-varying drone ground speed model based on wind speed and direction is further established to characterize the influence of meteorological conditions on drone flight efficiency. An enhanced adaptive large neighborhood search algorithm is subsequently proposed. Collectively, these contributions render the proposed model particularly well suited to the complex and dynamic scenarios associated with urban waterlogging during heavy rainfall events.

Table 1 summarizes relevant research on drone-truck collaboration for relief supply delivery, providing a comparative overview with this study.

Table 1. Research overview.

Reference	Objective	Dynamic conditions	Transport mode	Solution approach
Dukkanci et al. (2023)	Minimize the total unsatisfied demand	×	Large and small drones	Scenario Decomposition Algorithm
Yang et al. (2025)	Minimize the total travel time	×	Trucks and drones	Enhanced Two-Stage Heuristic Algorithm Based on Parallel Computing
Peng et al. (2025)	Minimize travel costs, delay penalties, and ensure fairness	√	Trucks and drones	MG, IAM-PER-MADDPG
Faiz et al. (2024)	Minimize cost	×	Hotspot drones and cargo drones	Benders Decomposition
Huang et al. (2022)	Minimize cost	×	Trucks and drones	Ant Colony Optimization
Ghelichi et al. (2021)	Minimize the total completion time	×	Drones	Time-Slot Formulation with Reduction Method
Wu et al. (2022)	Minimize delivery time	×	Trucks and drones	Improved Variable Neighborhood Descent
Gong et al. (2025)	Minimize total delivery time	√	Trucks, speedboats, and drones	Improved Adaptive Large Neighborhood Search

Continued on next page

Reference	Objective	Dynamic conditions	Transport mode	Solution approach
Zhou et al. (2026)	Maximize task assignment rate, prioritizing high-value tasks	√	Rescue vessels	Priority-aware Distributed Proximal Policy Optimization
Cheng et al. (2024)	Minimize total delivery time	√	Trucks, drones and motorcycles	Improved Adaptive Large Neighborhood Search
Li et al. (2025)	Maximize profit and reduce delivery delays	√	Trucks and drones	Resilient Adaptive Large Neighborhood Search
Shi et al. (2026)	Minimize waiting time and road risk	√	Hybrid fleet of trucks and drones	Q-learning and NSGA-II Hybrid Algorithm
Luo et al. (2021)	Minimize total completion	×	Trucks and drones	Multi-start Tabu Search Algorithm and Critical Path Method
Ribeiro et al. (2022)	Minimize drone use, charging station use, and drone route cost	×	Drones and mobile charging stations	Construct-and-adjust Heuristic Method integrated with a Genetic Algorithm
This study	Minimize the total rescue time, total cost, and average penalty	√	Emergency vehicle and drones	EALNS

3. Problem description and model formulation

3.1. Problem description

In this paper, we investigate the optimization of rescue routes through coordinated operations between urban emergency vehicles and drones in response to urban emergencies caused by torrential rain and flooding. Under conditions of extreme rainfall and inundation, urban roads are susceptible to waterlogging and congestion, with traffic conditions fluctuating in real time according to rainfall intensity and water depth. These conditions impede the timely delivery of relief supplies, severely constraining emergency response efficiency.

To address this challenge, we propose a collaborative rescue framework integrating emergency vehicles and drones. Through division of labor and complementary functions, both transport modes jointly accomplish rescue missions. Compared with conventional emergency vehicles, drones offer superior maneuverability and flexibility, enabling rapid response capabilities. Even when road networks are compromised, drones can sustain basic relief supply delivery via aerial corridors, thereby enhancing the robustness and disaster resilience of emergency systems. We specify two modes of relief supply transportation: First, emergency vehicles transport supplies to distribution centers, from which drones subsequently deliver them to severely flooded or vehicle-inaccessible locations, enabling precise and rapid last-mile delivery; second, emergency vehicles deliver supplies directly to designated locations.

We assume that the depot with sufficient supplies is situated at fixed urban locations and equipped with multiple emergency vehicles, while distribution centers accommodate multiple drones.

Emergency vehicles depart from the depot, transport supplies to distribution centers, and subsequently return to the center for additional tasks. Alternatively, when road conditions permit, vehicles deliver supplies directly to demand locations. Demand points are distributed across the urban road network, each characterized by specific supply requirements, soft time window constraints, and urgency levels. The proposed delivery system is shown in Figure 1.

Within dynamic urban flooding scenarios, vehicle speeds across road segments vary as time-dependent functions influenced by water depth. In contrast, the ground speed of drones is dynamically influenced by wind conditions, while their take-off and landing are contingent upon the current location of the distribution center, and are further constrained by battery endurance. We integrate the total operational costs, temporal efficiency, and penalty costs of the rescue system to construct a multi-objective optimization model that facilitates coordinated decision-making regarding vehicle and drone routing, service allocation, and temporal scheduling.

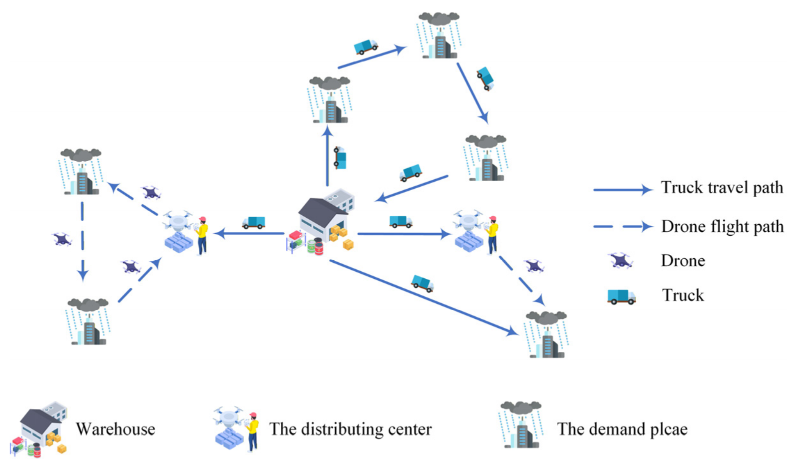


Figure 1. Schematic diagram of the vehicle-drone collaborative system.

3.2. *Problem hypothesis*

Based on the operational characteristics of emergency vehicles and drones under heavy rainfall disaster conditions, we establish the following five fundamental assumptions: (1) Drones may take off and land only at the distribution center, and each sortie must satisfy constraints on endurance and payload capacity. (2) The travel speed of emergency vehicles on road segments is governed by the depth of road inundation. (3) The flight speed of drones varies dynamically in response to prevailing meteorological conditions. (4) All emergency vehicles and drones employed are of a homogeneous type. (5) Each demand point is subject to a time window constraint within which relief supplies must be delivered.

3.3. Model formulation

In this subsection, we present the sets, parameters, decision variables, and constraints employed in the model formulation. The notation is summarized in Table 2.

Table 2. Notation for the model formulation.

Sets and parameters	Description
Sets	
D	Set of demand locations
V	Set of demand locations, distribution centers, and the depot, i.e., $V = D \cup H \cup \{0\}$
K	Set of emergency vehicles
U	Set of drones
H	Set of distribution centers
$\Omega = \{1,2,\cdots,\|T\|\}$	Set of weather time window indices
$T = \{T^1, T^2, \dots, T^{\|T\|}\}$	Set of weather time windows
Parameters	
q_i	Demand quantity at demand location i
c^k	Unit distance transport cost for vehicle k
c_f^k	Fixed deployment cost of vehicle k
c^u	Unit distance transport cost for drone u
c_f^u	Fixed deployment cost of drone u
M	A sufficiently large positive constant
$[e_i, l_i]$	Soft time window for demand location i
Q_k	Maximum loading capacity of vehicle k
W_u	Maximum payload capacity of drone u
s_i	Unloading and rescue operation time at demand location i
w_i	Waiting time at the demand location i
E_u	Maximum endurance time of drone u
d_{ij}	Distance between nodes i and j
$v_{ij}^{(m)}$	Average speed of vehicle k under waterlogging conditions
v_u	Flight speed of drone u
δ_i	Urgency level of demand location i
d_p	Maximum traversable waterlogging depth for vehicle k
$T^m = [w^m, w^{m+1}]$	Start and end times of the m-th time window
v_a	Airspeed of the drone
$v_w^{(m)}$	Wind speed during time window T^m
θ_m	Wind direction angle during time window T^m
∂_{hj}	Track angle of the drone from h to j (ground speed vector angle)
$\varphi_{hj}^{(m)}$	Heading angle during time window T^m (airspeed vector angle)
$vg_{hj}^{(m)}$	Drone ground speed on arc (h,j) during time window T^m
$t_{hj}^u(m)$	Drone flight time on arc (h,j) during time window T^m
$\rho_{hj}^{(m)}$	Risk probability for the drone traversing arc (h,j) during time window T^m
$\bar{R}$	Maximum cumulative risk threshold for the drone
P_u	Energy consumption of the drone per unit time

Continued on next page

Sets and parameters	Description
Decision variables	
x_{ij}^k	Whether vehicle k travels from node i to node j
y_{hj}^u	Whether drone u flies from distribution centre h to demand location j
a_i	Arrival time of vehicle or drone at demand location i
$P_i(a_i)$	Penalty cost incurred if vehicle or drone arrives at demand location i at time a_i
t_{ij}^k	Travel time for vehicle traversing from node i to node j
t_{hj}^u	Flight time for drone travelling from node h to node j
v_{hjm}^{u}	Binary variable equal to 1 if drone u traverses arc (h,j) during time window T^m, and 0 otherwise
b_j^u	Remaining battery level of drone u upon arrival at node j
ψ_j^u	Cumulative risk probability of drone u upon arrival at node j

In this study, the JMC Kaiyun box-type light truck is adopted as the rescue vehicle, with a maximum payload capacity of approximately 1.5 tons, while the DJI FlyCart 100 is selected as the drone, with a payload capacity of 80 kg. If emergency supplies consist of compressed biscuits weighing approximately 30 g per unit, the maximum capacities for emergency vehicles and drones are set at 50,000 and 2,500 units, respectively.

Given the critical importance of timely relief supply delivery to affected locations, we define $P_i(a_i)$ as the time window deviation penalty incurred when a vehicle or drone arrives at demand location i at time a_i.The penalty function is defined as follows:

$$P_i(a_i)=\begin{cases}\dfrac{e_i-a_i}{l_i-e_i}, a_i<e_i\\ 0 \qquad\quad , e_i<a_i<l_i\\ \dfrac{a_i-l_i}{l_i-e_i}, a_i>l_i\end{cases} \tag{1}$$

In this section, we formulate a multi-objective cooperative scheduling problem for vehicles and drones that incorporates emergency response time, vehicle operational costs, and penalty costs. The model is formulated as follows:

The decision variables comprise two categories: vehicle routing variables x_{ij}^k and drone flight route variables y_{ju}^h, defined as follows:

$$x_{ij}^k=\begin{cases}1, & if\ vehicle\ k\ travels\ from\ node\ i\ to\ j\\ 0, & otherwise\end{cases} \tag{2}$$

$$y_{hj}^u=\begin{cases}1, & if\ drone\ u\ flies\ from\ node\ h\ to\ j\\ 0, & otherwise\end{cases} \tag{3}$$

Given the significant temporal variability of rainfall processes, the travel time required for emergency vehicles to traverse the same road segment fluctuates over time. To accurately estimate the real-time travel duration of emergency vehicles during prolonged heavy rainfall, we calculate vehicle speeds across road segments based on simulated road flooding data, thereby constructing a dynamic speed matrix.

Du and Yang [61] proposed a speed decay model relating vehicle speed to water depth under extreme rainfall conditions, which enables calculation of the average travel speed of emergency vehicles on flooded road segments. If the water depth exceeds the maximum traversable depth for vehicles, the travel speed for that segment is set to zero, indicating that the vehicle cannot proceed. Let δ (min) denote the time step for urban road water depth simulation. The average travel speed of emergency vehicles can be derived for each time interval based on the water depth data at each simulation time point. The formulation is presented in Equation (4):

$$v_{ij}^{(m)} = \begin{cases} \left[\frac{v_0}{2} \cdot \tanh\left(\frac{-x_{ij}^{(m)} + 0.5 d_p}{3} \right) + \frac{v_0}{2} \right] \Big/ 3.6, & x_{ij}^{(m)} < d_p \\ 0, & x_{ij}^{(m)} \geq d_p \end{cases} \tag{4}$$

In the above equation, v_0 denotes the design speed of the road under normal conditions without flooding (km/h); d_p represents the maximum traversable water depth for emergency vehicles (cm); $v_{ij}^{(m)}$denotes the vehicle speed on the road segment at the m-th simulation time step (m/s); and $x_{ij}^{(m)}$ denotes the actual water depth on the road segment at the corresponding time step (cm).

In rainstorm and waterlogging disaster scenarios, drone flight performance is significantly affected by time-varying weather conditions. Unlike ground vehicles, drones are directly exposed to the atmospheric environment, with flight efficiency and safety subject to the influence of rainfall intensity and wind velocity and direction. We develop a time-varying impact model for drone flight under rainstorm conditions from two perspectives: delivery efficiency and flight safety.

Based on the phased characteristics of the rainstorm process, the rescue period is divided into multiple weather time windows. Let $T = \{T^1, T^2, \dots, T^{|T|}\}$ denote the set of weather time windows, and $\Omega = \{1,2, \dots, | T |\}$ denote the set of time window indices. The m-th time window is expressed as $T^m = [w^m, w^{(m+1)}]$, where w^{m} and $w^{(\mathrm{m}+1)}$ represent the start and end times, respectively. Within each time window, weather conditions (wind velocity, wind direction, and rainfall intensity) are assumed to remain constant. Based on the classification outlined above, the model developed in this study is essentially a continuous-time routing problem with time-dependent parameters. The flight time and risk probability of a drone on arc (h, j) are not static inputs; rather, they depend on the meteorological window T^m within which the departure time a_h falls. To this end, a binary variable v_{hjm}^u is introduced to encode this correspondence, ensuring that each flight mission invokes the wind speed, wind direction, and risk parameters associated with the corresponding window.

During flight, the airspeed v_a of a drone is affected by wind velocity and direction, which in turn influences the actual flight time. The drone performs its mission based on ground speed. Let $v_w^{(m)}$ denote the wind speed during time window T^m, and θ_m denote the wind direction angle. When the drone travels from node h to node j, the track angle (ground speed vector angle) is ∂_{hj}. Following Thibbotuwawa et al. [62], the heading angle (airspeed vector angle) $\varphi_{hj}^{(m)}$ is calculated as follows:

$$\varphi_{hj}^{(m)} = \partial_{hj} - \arcsin\left(\frac{v_w^{(m)}}{v_a} \sin(\theta_m - \partial_{hj})\right) \quad \forall m \in \Omega, (h, j) \in V \tag{5}$$

The ground speed $vg_{hj}^{(m)}$ of the drone traveling from node h to node j during time window T^m is obtained by combining the airspeed vector and the wind velocity vector:

$$vg_{hj}^{(m)} = \sqrt{(v_a \times \cos\varphi_{hj}^{(m)} + v_w^{(m)} \times \cos\theta_m)^2 + (v_a \times \sin\varphi_{hj}^{(m)} + v_w^{(m)} \times \sin\theta_m)^2} \quad \forall m \in \Omega, (h, j) \in V \tag{6}$$

Accordingly, the flight time for the drone traveling from node h to node j during time window T^m is given by:

$$t_{hj}^{u}(m) = d_{hj} \Big/ vg_{hj}^{(m)} \quad \forall m \in \Omega, (h, j) \in V \tag{7}$$

Rainstorm conditions increase the operational risks for drones. Heavy precipitation may cause short circuits in the drone's electronic components, while strong crosswinds can compromise flight stability. Let $\rho_{hj}^{(m)}$ denote the risk probability faced by a drone traveling from node h to node j during time window T^m. During mission execution, the cumulative risk probability across all visited nodes must not exceed the predefined risk threshold $\bar{R}$. To this end, a binary variable v_{hjm}^{u} is introduced, which equals 1 if drone u travels from node h to node j during time window T^m, and 0 otherwise.

To account for response time, response cost, and demand-related service penalties in emergency rescue operations, the objective function is formulated as follows:

Objective 1: Minimize Total Response Time

$$\min f_1 = \sum_{k \in K} \sum_{(i,j) \in V} t_{ij}^{k} \cdot x_{ij}^{k} + \sum_{u \in U} \sum_{h \in H, j \in D} t_{ij}^{u} \cdot y_{hj}^{u} + \sum_{j \in D} s_j \cdot (\sum_{k \in K} x_{ij}^{k} + \sum_{u \in U} y_{hj}^{u}) \tag{8}$$

In the above equation, t_{ij}^{k} denotes the travel time for vehicle k from node i to node j; t_{hj}^{u} denotes the flight time for drone u from node h to node j; and s_j denotes the unloading and rescue operation time at demand location j.

Objective 2: Minimize Total Response Cost

$$\min f_2 = \sum_{k \in K} c^k \sum_{i \in V} \sum_{j \in V} d_{ij} x^k{}_{ij} + \sum_{k \in K} c_k^f \sum_{j \in D \cup H} x^k{}_{0j} + \sum_{u \in U} c^u \sum_{h \in H} \sum_{j \in D} d_{ij} y^u{}_{hj} + \sum_{u \in U} c_u^f \sum_{j \in D} y_{hj}^{u} \tag{9}$$

In the above equation, c^k denotes the unit distance transport cost for vehicle k; c^u denotes the unit distance transport cost for drone u; c^f denotes the fixed deployment cost of vehicle k; and d_{ij} denotes the distance between node i and demand location j. Although total response time and total response cost are influenced by routing and resource-allocation decisions, they capture distinct and non-interchangeable performance dimensions. Total response time, as defined in Eq. (10), measures the timeliness of the rescue operation by accounting for vehicle travel time, drone flight time, and on-site unloading and rescue time. By contrast, total response cost, as defined in Eq. (11), reflects the economic resources required for the operation, including distance-dependent transportation costs and fixed vehicle deployment costs. The two objectives are not necessarily proportional. In resource deployment decisions, deploying additional vehicles or drones can shorten response time but increase fixed and variable costs, whereas minimizing cost alone may reduce deployed resources and prolong response time. Retaining both objectives therefore enables the model to capture the trade-off between rapid emergency response and cost-efficient resource utilization. Moreover, the multi-objective formulation avoids imposing an arbitrary conversion between time and monetary cost and provides decision-makers with Pareto-efficient alternatives under different operational priorities.

Objective 3: Minimize Average Demand Penalty

$$\min f_3 = \frac{1}{h} P_i\left(a_i\right) \tag{10}$$

In this formulation, a_i denotes the arrival time of the rescue vehicle at demand i. The term $P_i(ai)$ represents the time window deviation penalty incurred at demand i when the vehicle arrives at time a_i. This penalty does not constitute a direct economic cost; rather, it serves as a quantitative measure of rescue service timeliness. Specifically, if the rescue vehicle or drone arrives before the earliest permissible service time e_i, a waiting penalty is imposed; conversely, arrival after the latest permissible service time l_i incurs a delay penalty. Incorporating penalty minimization as an independent objective explicitly captures the time-sensitivity requirements inherent in rescue missions, preventing such considerations from being subsumed by the aggregate cost objective. This approach refines the decision-making mechanism of the optimization model: Rather than solely pursuing minimum economic cost, the model seeks to maximize rescue timeliness within acceptable cost bounds.

Subject to:

$$\sum_{j \in D} x_{0j}^k - \sum_{i \in D} x_{i0}^k = 0 \quad \forall k \in K \tag{11}$$

$$\sum_{i \in V} x_{ij}^k = \sum_{i \in V} x_{ji}^k \quad \forall j \in V, k \in K \tag{12}$$

$$\sum_{j \in D} q_j x_{ij}^k < Q_k \quad \forall i \in V, k \in K \tag{13}$$

$$s_0 = w_0 = 0 \tag{14}$$

$$w_i = \max\left\{0, e_i - a_i\right\} \quad \forall i \in D \tag{15}$$

$$\sum_{j \in D} q_j y_{hj}^u < W_u \quad \forall h \in H, \forall u \in U \tag{16}$$

$$y_{hj}^u = \sum_{m \in \Omega} v_{hjm}^u, \quad \forall u \in U, h \in H, j \in D \tag{17}$$

$$\psi_h^u = 0, \quad \forall h \in H, u \in U \tag{18}$$

$$\psi_j^u \leq 1 - (1 - \psi_h^u)(1 - \rho_{hj}^{(m)}) + M(1 - v_{hjm}^u), \quad \forall h \in H, j \in D, u \in U, m \in \Omega \tag{19}$$

$$\psi_j^u \geq 1 - (1 - \psi_h^u)(1 - \rho_{hj}^{(m)}) - M(1 - v_{hjm}^u), \quad \forall h \in H, j \in D, u \in U, m \in \Omega \tag{20}$$

$$\psi_j^u \le \overline{R} \cdot \sum_{h \in H} y_{hj}^u, \quad \forall j \in D, u \in U \tag{21}$$

$$b_j^u \le b_h^u - P_u \cdot t_{hj}^u(m) + M(1 - v_{hjm}^u), \quad \forall h \in H, j \in D, u \in U, m \in \Omega \tag{22}$$

$$b_j^u \ge P_u \cdot t_{jh}^u(m), \quad \forall h \in H, j \in D, u \in U, m \in \Omega \tag{23}$$

$$\sum_{j \in D} \sum_{m \in \Omega} t_{hj}^u(m) v_{hjm}^u + \sum_{j \in D} \sum_{m \in \Omega} t_{jh}^u(m) v_{jhm}^u \le E_u \quad \forall h \in H, u \in U \tag{24}$$

$$\sum_{u \in U} \sum_{j \in D} y_{hj}^u \le 1 \quad \forall h \in H \tag{25}$$

$$y_{hj}^u \le \sum_{k \in K} \sum_{i \in V} x_{ih}^k \quad \forall h \in H, i \in D, u \in U \tag{26}$$

$$\sum_{k \in K} \sum_{i \in V} x_{ij}^k + \sum_{u \in U} \sum_{h \in H} y_{hj}^u = 1 \quad \forall j \in D \tag{27}$$

$$a_h = \sum_{k \in K} \sum_{i \in V} x_{ih}^k \left(a_i + s_h + w_h + t_{ih}^k \right) \quad \forall h \in H \tag{28}$$

$$a_j = \sum_{k \in K} \sum_{i \in V} x_{ij}^k \left(a_i + s_j + w_j + t_{ij}^k \right) + \sum_{u \in U} \sum_{h \in H} y_{hj}^u (a_h + s_j + w_j + t_{hj}^u) \quad \forall j \in D \tag{29}$$

$$\sum_{h \in H} y_{hj}^u = \sum_{h \in H} y_{jh}^u \quad \forall j \in D, \forall u \in U \tag{30}$$

$$Q_k \sum_{i \in V} x_{ij}^k + W_u \sum_{h \in H} y_{hj}^u \ge q_j \quad \forall k \in K, u \in U, j \in D \tag{31}$$

$$a_i \ge 0 \quad \forall i \in V \tag{32}$$

In the above mathematical model, let D denote the set of demand point vertices, and V denote the set comprising demand points, distribution centers, and the depot, defined as $V = D \cup H \cup \{0\}$. The set of emergency vehicles is denoted by K. Constraint (11) ensures that each vehicle route originates from and returns to the depot. Constraint (12) enforces flow conservation along vehicle routes. Constraint (13) ensures that the demand at each demand point does not exceed the capacity of any emergency vehicle, where Q_k denotes the capacity of vehicle k. Constraint (14) defines the waiting and service times at the depot, where s_0 and w_0 denote the service and waiting times at the depot, respectively. Constraint (15) defines the waiting time. Constraint (16) ensures that the demand at each demand point does not exceed the capacity of any drone, where W_u denotes the capacity of drone u. Constraint (17) defines the relationship between the arc selection variable and the time window traversal variable. Constraint (18) initializes the cumulative risk at the distribution center to zero. Constraints (19)–(20) govern cumulative risk propagation and threshold enforcement, where m $\in \Omega$ denotes the index of the meteorological time window, ensuring that the cumulative risk probability during drone mission execution does not exceed the predefined threshold $\bar{R}$. Constraint (21) tracks the depletion of the drone's remaining battery level. Constraint (22) ensures that the drone retains sufficient battery upon arrival at any demand node to return to the distribution center. Constraints (23)–(24) ensure that the total flight time of each drone does not exceed its maximum endurance E_u. Constraint (25) specifies that each distribution center h may launch at most one drone per dispatch cycle. Constraint (26) stipulates that drones may launch only from distribution centers previously visited by a vehicle. Constraint (27) ensures that each demand j is served exactly once, either by a vehicle or by drone. Constraints (28) and (29) establish the relationship between arrival times at demand points i and j, where t_{ij} denotes the travel time from node i to demand point j. Constraint (30) enforces flow conservation along drone routes. Constraint (31) requires that the total quantity of relief supplies delivered by drones and vehicles to each demand point must meet or exceed the demand at that location.

3.4. Model reformulation

In multi-objective optimization problems, inherent conflicts among objectives often render the direct computation of Pareto frontiers highly complex. To address this issue, we adopt the ε-constraint method to generate a set of nondominated solutions for the proposed multi-objective model. Rather than solving the problem only once as a single-objective model, the ε-constraint method constructs a family of scalar subproblems by retaining one objective as the primary objective and imposing upper bounds on the remaining objectives. By solving these subproblems under different ε combinations and applying Pareto dominance filtering, an approximated Pareto frontier can be obtained. The original model comprises three objective functions: minimization of total rescue time, total rescue costs, and average demand-point penalties.

Given the urgency of emergency rescue operations during torrential rain and urban flooding, temporal efficiency constitutes the primary concern. Accordingly, we designate the minimization of total rescue time as the principal objective, while the remaining two objectives, namely the minimization of average demand-point penalties and the minimization of total rescue cost, are treated as constrained objectives bounded by ε parameters. Since the ε-constraint method directly converts secondary objectives into constraints, eliminating the need for weighted aggregation of objectives with disparate units, it inherently avoids the bias introduced by subjective weight assignment. This formulation prioritizes temporal efficiency while enabling the trade-offs among response time, response cost, and service timeliness to be examined through different ε combinations. The resulting objective vectors are then collected and filtered according to the Pareto dominance criterion, so that the trade-offs among response time, response cost, and service timeliness can be represented by a nondominated solution set. The reconstructed constraints are presented as follows:

$$\sum_{k\in K} c^k \sum_{i\in V}\sum_{j\in V} d_{ij} x_{ij}^k + \sum_{k\in K} c_k^f \sum_{j\in D\cup H} x_{0j}^k + \sum_{k\in K} c^u \sum_{h\in H}\sum_{j\in D} d_{ij} y_{hj}^u + \sum_{u\in U} c_u^f \sum_{j\in D} y_{hj}^u \le \varepsilon_1 \tag{33}$$

$$\frac{1}{h} P_i(a_i) \le \varepsilon_2 \tag{34}$$

Here, ε_1 and ε_2 denote the permissible upper bounds for total rescue costs and average demand-point penalties, respectively. To appropriately set the constraint thresholds, the feasible range for ε need to be established using the payoff matrix. An approximate nondominated solution set is generated by solving multiple ε-constrained subproblems, each defined by a different pair of ε values.

This is accomplished by solving three single-objective optimization problems to obtain the optimal value f_i^U and the corresponding optimal solution x_i^* for each objective function. Each single-objective optimal solution x_i^* is subsequently substituted into the remaining objective functions to calculate the corresponding objective values, thereby forming the payoff matrix Φ, as presented below.

$$\Phi = \begin{bmatrix} f_1(x_1^*) & f_2(x_1^*) & f_3(x_1^*) \\ f_1(x_2^*) & f_2(x_2^*) & f_3(x_2^*) \\ f_1(x_3^*) & f_2(x_3^*) & f_3(x_3^*) \end{bmatrix} \tag{35}$$

From this matrix, the best and worst values for each objective can be derived, denoted as f_i^U and f_i^{SN}, respectively. To ensure uniform distribution of generated solutions, the grid method is employed to sample ε_1 and ε_2 at regular intervals within the established range. For each secondary objective f_i, a grid partition number m_{ij} is specified, and the constraint values are generated according to the following formula:

$$\varepsilon_{ij} = f_i^{SN} - \frac{\left(f_i^{SN} - f_i^U\right)}{m_{ij}} \cdot j \qquad j = 1, 2, \ldots m_{i,\max} \tag{36}$$

After all the ε-constrained subproblems have been solved, any infeasible or duplicate solutions are removed. The remaining feasible solutions are then examined using the Pareto dominance criterion: For a minimization problem, a solution x_i is said to dominate another solution x_j if $f_k(x_i) \leq f_k(x_j)$ for all objectives $k = 1,2,3,$ and $f_k(x_i) < f_k(x_j)$ for at least one objective. Any solution that is dominated by at least one other feasible solution is discarded. The solutions that remain after this filtering constitute the nondominated solution set obtained from the ε-constraint method. Since the ε values are sampled using a finite grid, this set should be regarded as an approximation of the Pareto frontier rather than a complete enumeration of all possible nondominated solutions.

To identify the compromise solution closest to the ideal point from the generated Pareto front, we employ an ideal-point-based compromise solution selection method. The underlying rationale is to determine an optimal compromise from among the Pareto-efficient solutions obtained via the ε-constraint method, guided by the principle of overall optimality. Specifically, an ideal point is constructed from the diagonal elements of the payoff matrix, representing the theoretical optimum achievable when each of the three objectives is optimized independently. The objective values of each Pareto solution are then normalized on a relative basis, and the solution exhibiting the minimum distance to the ideal point is selected as the compromise solution.

4. Solution approach

In this study, we address the multi-objective vehicle-drone collaborative route optimization problem in dynamic environments, which is NP-hard. The objective function incorporates a dynamic speed function influenced by water depth and soft time window penalties, rendering large-scale instances intractable for traditional exact algorithms and commercial solvers such as CPLEX within reasonable computational time. Adaptive Large Neighborhood Search (ALNS) is a metaheuristic algorithm widely applied to complex combinatorial optimization problems. Through the synergistic interaction of destroy and repair mechanisms, ALNS effectively prevents premature convergence to local optima, offering notable advantages in search efficiency and convergence performance. The EALNS algorithm proposed in this study is built upon a standard ALNS framework, incorporating several enhancement mechanisms tailored to the dynamic environment of urban waterlogging under heavy rainfall conditions. By embedding disaster evolution dynamics into the operator design, the algorithm effectively addresses large-scale collaborative route planning problems in dynamic disaster environments. Specifically, two dedicated removal operators, namely the dynamic inundation-aware removal operator and the predictive inundation-aware removal operator, integrate real-time flood prediction data from the Storm Water Management Model (SWMM) into the search process, thereby enhancing the algorithm's adaptability to time-varying inundated road networks. The vehicle–drone collaborative insertion operator and the predictive collaborative insertion operator dynamically balance two competing repair strategies: Reassigning the task to a drone or completing delivery via vehicle prior to road closure. Flight risk thresholds are incorporated into the repair logic to account for the time-varying characteristics of meteorological conditions. Furthermore, the adaptive disruption magnitude mechanism dynamically adjusts disruption intensity based on search history: When the search stagnates in a local optimum, the disruption range is expanded to promote global exploration; when sustained improvement is observed, disruption intensity is reduced to reinforce local search. Although this mechanism bears resemblance to local search strategies in existing ALNS variants, the integration of real-time inundation prediction data confers substantially greater dynamic adaptability. Collectively, these enhancements ensure the efficiency and robustness of the EALNS algorithm in complex urban waterlogging scenarios induced by heavy rainfall.

During initialization, a high-quality feasible solution is first constructed, with all destroy and repair operators assigned an initial weight of 1. In subsequent iterations, a pair of removal and insertion operators is selected via roulette wheel selection based on current operator weights. The removal operator partially destroys the current solution, while the insertion operator performs repair and optimization operations to generate a new one. The new solution is evaluated against the simulated annealing acceptance criterion to determine whether it replaces the current solution. Concurrently, operator weights are dynamically adjusted based on their performance throughout the optimization process. These iterative steps continue until the termination criteria are satisfied, yielding the optimized solution. The overall algorithmic workflow is illustrated in Figure 2.

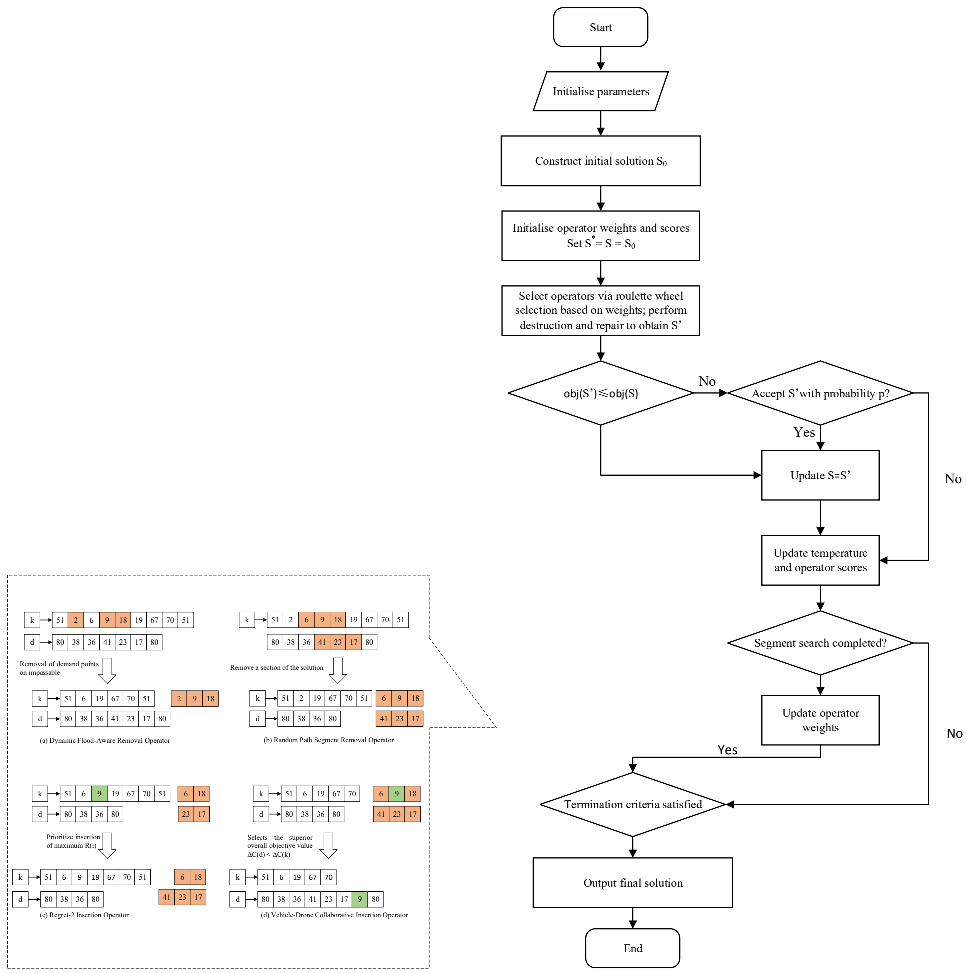


Figure 2. Flowchart of the EALNS algorithm and partial operators.

4.1. Initial solution generation

The initial solution serves as the starting point for algorithmic iteration, and its quality significantly influences the subsequent optimization process. A high-quality initial solution facilitates faster convergence and enhances the performance of the final solution. We employ a method based on service mode classification and collaborative route construction to generate initial solutions.

Specifically, demand locations are first categorized into two service types based on real-time inundation depth, time window urgency, and geographical position, namely direct vehicle service and drone service. The classification criteria are as follows: Demand locations where the current water depth $x_{ij}^{(m)}$ on the associated road segment exceeds the safety threshold d_{safe}, or where water depth is predicted to surpass the critical value d_p during the rescue period, are prioritized for drone service; all remaining demand locations are assigned to direct emergency vehicle service. For demand locations assigned to direct vehicle service, a greedy insertion strategy based on composite urgency is employed. First, the composite urgency score U_i is calculated for each demand location, and locations are ranked in descending order of urgency. Each location is then sequentially inserted into the current vehicle route at the position incurring the minimum additional cost, while ensuring that the vehicle capacity constraint Q_k and the time window constraint $[e_i, l_i]$ are satisfied.

$$U_i = \omega_1 \frac{1}{(l_i - e_i)} + \omega_2 \delta_i \tag{37}$$

For demand locations assigned to drone relay service, a two-stage allocation strategy is employed. First, the K-means clustering algorithm is applied based on Euclidean distance to assign demand locations to the nearest distribution center $h \in H$, forming multiple drone service clusters. Subsequently, within each cluster, drone flight routes are constructed using the nearest neighbor method, subject to the drone endurance constraint E_u and capacity constraint W_u. Upon completion of service allocation, distribution center visits are integrated into emergency vehicle routes. For each distribution center requiring drone service, a transport task from the depot to that center is generated. The cost increment of inserting this task at each candidate position along existing vehicle routes is then evaluated, and the feasible position yielding the minimum total cost increase is selected.

Although the method based on service mode classification and collaborative route construction can generate feasible initial solutions, the quality of solutions produced by a single heuristic may vary considerably across problem instances. To ensure that the algorithm commences from high-quality solutions, we design an initial solution quality assurance mechanism based on multi-strategy parallel evaluation. This mechanism employs three complementary construction strategies in parallel: (1) Urgency-priority strategy: This strategy ranks all demand points in descending order of urgency δ_i and demand quantity q_i, prioritizing resource allocation to points with high urgency and great demand. Demand points are sequentially inserted into existing vehicle routes or assigned to newly created routes. This strategy ensures that time-critical rescue tasks receive priority response. (2) Distance-priority strategy: This strategy employs a nearest neighbor heuristic for route construction. At each iteration, the demand point nearest to the current location that satisfies capacity constraints is selected and appended to the route until vehicle capacity is exhausted. This strategy aims to minimize total travel distance for vehicles and drones, thereby reducing delivery costs. (3) Random construction strategy: This strategy randomly selects unserved demand points for route construction, ensuring diversity in initial solutions. This approach avoids the local structural traps into which greedy strategies may fall, thereby providing a larger search space for subsequent iterative optimization.

The algorithm employs the three strategies described above to generate initial solutions S_1, S_2, and S_3, evaluates their respective objective function values, and selects the solution with the minimum objective value as the starting point for EALNS iteration:

$$S_0 = \arg\min\{f(S_1), f(S_2), f(S_3)\} \tag{38}$$

This mechanism reduces the sensitivity of algorithmic performance to initial solution quality, enabling the algorithm to adapt robustly to diverse problem characteristics.

4.2. Destroy and repair mechanisms

In the ALNS algorithm, destroy operators and repair operators collectively form the core search mechanism to progressively improve the current solution. Destroy operators remove selected task nodes from the existing solution to create an incomplete intermediate solution, while repair operators reinsert the removed nodes to generate a new solution. Extending classical operators and tailoring them to the specific characteristics of the problem under investigation, we design four categories of destroy operators and four categories of repair operators. The schematic diagrams of partial operators are illustrated in Figure 2. The specific descriptions are as follows:

4.2.1. Destroy operator

The destroy operations within the EALNS algorithm are designed to enhance solution diversity and global search capability, particularly for addressing dynamic road conditions during heavy rainfall-induced urban flooding, thereby enabling the algorithm to escape from local optima. Tailored to the characteristics of emergency collaborative delivery problems, we employ multiple destroy operators, including random removal, worst-cost removal, time-window-related removal, and route-segment removal, to disrupt current solutions across dimensions, thereby enhancing algorithmic adaptability and search efficiency under dynamic flooding scenarios. Four operators are introduced: the dynamic flood-aware removal operator, the Shaw removal operator, the worst-cost removal operator, and the random route-segment removal operator.

(1) Dynamic Flood-Aware Removal Operator: This operator prioritizes the removal of demand points located on road segments that are currently impassable or predicted to become impassable, based on real-time road trafficability data derived from the flood evolution model.

(2) Shaw Removal Operator: The Shaw removal operator enhances solution diversity by removing clusters of highly similar demand points, thereby strengthening the algorithm's global search capability in dynamic environments. The operator first randomly selects a demand point as the seed node, then evaluates pairwise node similarity across dimensions, including spatial distance, time window, and demand urgency, before removing a group of similar nodes. The similarity measure $r(i, j)$ is defined as follows:

$$r(i,j)=\omega_1\frac{d_{ij}}{d_{\max}}+\omega_2\frac{\left|(e_i+l_i)/2-(e_i+l_i)/2\right|}{\tau_{\max}}+\omega_3\frac{\left|\delta_i+\delta_j\right|}{\delta_{\max}} \tag{39}$$

A smaller value of $r(i, j)$ indicates greater similarity between demand points i and j. Here, d_{ij} denotes the Euclidean distance between demand points i and j; $(e_i+l_i)/2$ represents the midpoint of the time window for demand point i; and δ_i denotes the demand urgency coefficient. The terms τ_{max} and δ_{max} represent the maximum values of each corresponding term, used for normalization. Parameters ω_1, ω_2, and ω_3 are weighting coefficients satisfying $\omega_1+\omega_2+\omega_3=1$.

(3) Worst-Cost Removal Operator: The core principle of this operator is to improve solution quality by removing the nodes that incur the highest costs. The execution proceeds in two steps: First, each node in the current route is evaluated to calculate the reduction in total route cost resulting from its removal; then, the nodes yielding the most significant cost reductions are selected and removed.

(4) Random Route Segment Removal Operator: This operator randomly selects a rescue route and designates a random node on that route as the starting point, then removes a contiguous segment of nodes beginning from that point to construct a partially disrupted intermediate solution. Unlike operators that remove isolated nodes in a scattered manner, this operator focuses on locally disrupting a single route. Compared to operators that remove entire routes, its disruption intensity is more moderate and controllable, thereby preserving the viable structure within the original route.

4.2.2. Repair operator

The primary function of repair operators is to reintegrate nodes removed by destroy operators into the current partial solution, thereby constructing a new complete feasible solution. In each iteration, the set of removed demand locations awaiting reassignment is progressively inserted into feasible vehicle or drone routes according to the selected repair strategy until all removed nodes are properly accommodated. To address the emergency collaborative routing problem under dynamic urban flooding conditions, we design four insertion operators: A greedy insertion operator, a Regret-2 insertion operator, a time-window urgency priority insertion operator, and a vehicle-drone collaborative insertion operator.

(1) Greedy Insertion Operator: In each iteration, this operator selects the node from the candidate set whose insertion causes the minimum increase in the objective function value and places it at the optimal position within the current solution. By evaluating the marginal cost associated with all feasible insertion positions for each node, the operator prioritizes the insertion scheme with the lowest marginal cost. This process continues until all removed nodes have been reinserted, progressively constructing an improved feasible solution while satisfying all constraints.

(2) Regret-2 Insertion Operator: This operator overcomes the limitations of local greedy strategies by evaluating the potential opportunity cost of inserting each candidate node at different positions. Its core mechanism involves calculating the regret value $R(i)$ for each demand node, defined as the difference in objective function increase between the suboptimal and optimal insertion positions, and prioritizing the insertion of nodes with the highest regret values.

(3) Time-Window Urgency Priority Insertion Operator: In emergency response to severe rainstorm flooding, the timely delivery of relief supplies to affected areas is critical. This operator prioritizes disaster-affected locations with the strictest time-window constraints or the most urgent demand, thereby minimizing the total system penalty cost and enhancing rescue timeliness. The urgency metric comprehensively considers time-window tightness and the inherent importance of each demand location. Candidate nodes are ranked by their time-window urgency, with the most critical nodes inserted first.

(4) Vehicle-Drone Collaborative Insertion Operator: The core principle of this operator is to simultaneously evaluate two service alternatives for each candidate demand location during the repair process: Direct service by vehicle and drone service dispatched from a nearby distribution center. The operator calculates the resulting changes in travel cost, time cost, and time-window penalties, then selects the alternative yielding the superior overall objective value.

4.3. Solution acceptance criterion

To prevent the algorithm from becoming trapped in local optima, we incorporate a Simulated Annealing (SA) mechanism into the EALNS algorithm as the solution acceptance criterion, balancing the algorithm's behavior between global exploration and local exploitation. The strategy operates as follows: In each iteration, a new solution S' is generated through removal and insertion operations and compared with solution S. Let $obj(S')$ and $obj(S)$ denote the objective function values of the new and current solutions, respectively. If the new solution is superior to the current one, $obj(S') < obj(S)$, then S' is accepted as the new solution. Furthermore, if the new solution also surpasses the historical best solution S^*, then S^* is updated to S'.

If the new solution is inferior to the current solution, acceptance is determined according to the simulated annealing probability criterion. The acceptance probability p is calculated as:

$$p = \exp[-(obj(S') - obj(S)) / T] \tag{40}$$

where T denotes the temperature parameter. The initial temperature T_{ini} is set based on the quality of the initial solution, typically calibrated such that a solution ω% worse than the initial solution has a 50% acceptance probability. During the iterative process, the temperature decreases gradually according to $T = T \times \varepsilon$, where ε denotes the cooling rate (0 < ε < 1). Given the complexity of the solution space and the numerous constraints inherent in this problem, ε is set to 0.97 to ensure sufficient time for global exploration and to avoid premature convergence to local optima, resulting in a relatively moderate rate of temperature decay. A higher cooling rate enables the algorithm to maintain a greater probability of accepting inferior solutions during the early search phase, thereby enhancing global search capability and facilitating escape from local optima. As iterations progress, the temperature gradually decreases and the probability of accepting inferior solutions diminishes correspondingly. This enables the algorithm to possess strong escape capability from local optima in the early stages while favoring fine-grained local search in the later stages, thereby progressively converging toward the global optimum.

This mechanism effectively prevents premature convergence, enhances global optimization capability, and ensures convergence and stability of the final solution.

4.4. Operator scoring mechanism

During the search process, to adaptively select effective operators, each destroy operator and repair operator is assigned a score based on its performance. A higher score corresponds to a greater probability of the operator being selected in subsequent iterations. At the beginning of each phase, the initial scores of all operators are set to zero. In each iteration, after a pair of destroy and repair operators generates a new solution, they are scored according to the quality of this solution: If the new solution represents a new global best, both operators receive σ_1 points; if the new solution improves upon the current solution but does not achieve global optimality, each receives σ_2 points; if the new solution is inferior yet accepted, each receives σ_3 points, where $\sigma_1 > \sigma_2 > \sigma_3$. Through this mechanism, the algorithm dynamically evaluates operator performance and prioritizes the selection of more effective operators.

4.5. Adaptive operator selection mechanism

Operator scores reflect only recent performance. To optimize the overall search strategy, these scores must be converted into long-term weights that guide operator selection. The EALNS algorithm incorporates a performance-based operator weight adjustment mechanism and a roulette wheel selection method to adaptively adjust the search strategy.

The algorithm divides the total iteration process into multiple consecutive segments. At the end of each segment, the weight of each operator is updated based on its cumulative score within that segment. The weight update formula is as follows:

$$\omega_i' = \lambda \cdot \omega_i + (1-\lambda) \cdot \frac{\pi_i}{\theta_i} \tag{41}$$

where ω_i' and ω_i denote the updated and previous weights of operator i, respectively, π_i represents the cumulative score obtained by the operator during the current segment according to the rules in Section 4.4, and θ_i denotes its invocation count. Parameter λ is the weight decay factor ($0 < \lambda < 1$), which balances the influence of historical weights and current performance. In emergency logistics problems under dynamically evolving severe rainstorm flooding, operator performance may fluctuate significantly across search phases due to time-varying road accessibility and complex constraints. A small λ value would cause weights to respond rapidly to individual performance outcomes, potentially leading to frequent weight fluctuations that compromise algorithmic stability. Therefore, a relatively large decay factor $\lambda = 0.65$ is adopted to better accumulate long-term operator performance, thereby avoiding weight misjudgement caused by short-term fluctuations and enhancing the algorithm's robustness in dynamic environments. At the beginning of each new segment, all operator scores π_i are reset to zero while their weights are retained, ensuring that operator selection is guided by long-term historical performance rather than short-term fluctuations and maintaining continuity in the search strategy.

Based on the updated weights, the algorithm employs a roulette wheel selection method to probabilistically select operators. The selection probability of each operator is proportional to its weight:

$$p_i = \frac{\omega_i}{\sum_{k=1}^{n} \omega_k} \tag{42}$$

where p_i denotes the selection probability of operator i, ω_i represents its current weight, and n is the total number of destroy or repair operators in the corresponding category. This mechanism ensures that operators with superior performance have higher selection probabilities, thereby guiding the search direction while preserving sufficient randomness to explore new possibilities.

4.6. Enhancement mechanisms

To further enhance algorithmic performance, we incorporate two enhancement mechanisms targeting local search capability and dynamic parameter adjustment. First, to improve route quality, a 2-opt local search operator is applied to vehicle and drone routes following initial solution generation. The 2-opt algorithm is a classical route improvement heuristic that reduces total travel distance by reversing subsequences within a route to eliminate crossings and reduce detours. For a route $R = \{r_0, r_1, \ldots, r_n\}$ containing n nodes, the 2-opt operator examines all pairs of edges (r_{i-1}, r_i) and (r_j, r_{j+1}). If reversing the sub-route $[r_i, \ldots, r_j]$ reduces the total distance, the reversal is accepted. The distance change resulting from reversing sub-route $[r_i, \ldots, r_j]$ is given by:

$$\Delta = d(r_{i-1}, r_j) + d(r_i, r_{j+1}) - d(r_{i-1}, r_i) - d(r_j, r_{j+1}) < 0 \tag{43}$$

Second, an adaptive destruction degree mechanism is introduced to dynamically adjust the removal rate ρ based on search history, thereby balancing global exploration and local exploitation.

When the algorithm fails to identify an improved solution over k consecutive iterations, the destruction degree is increased to enhance global search capability. Conversely, when improved solutions are consistently discovered, the destruction degree is reduced to intensify local exploitation. This adaptive mechanism enables the algorithm to employ stage-appropriate strategies throughout the search process: Larger destruction degrees facilitate escape from local optima during early iterations or stagnation periods, while smaller destruction degrees permit fine-grained search within promising regions as the algorithm converges.

4.7. Predictive search mechanism

In rainstorm and waterlogging scenarios, road trafficability changes dynamically over time. A passable road may become impassable within minutes due to rising water levels, forcing the interruption of an ongoing delivery route. To address this issue, a predictive search mechanism is introduced within the EALNS framework, comprising a predictive waterlogging-aware removal operator and a predictive collaborative insertion operator.

Predictive Waterlogging-Aware Removal Operator. The core idea of this operator is to identify and remove demand nodes that are "currently accessible but will soon become inaccessible", enabling them to be reassigned to better routes or switched to drone service during the repair phase. For a demand node j served by a vehicle, the algorithm first obtains the shortest path $P_j = \{e_1, e_2, \ldots, e_k\}$ from the depot to that node. Then, for each edge e_i along the path, the algorithm assesses whether it will become blocked within the prediction horizon Δt. The path blockage risk index for demand node j is defined as:

$$R_j = \frac{\left|\left\{e \in P_j : \hat{x}_e(t+\Delta t) > d_p\right\}\right|}{\left|P_j\right|} \tag{44}$$

where $\hat{x}_e(t + \Delta t)$ denotes the predicted water depth on edge e at time $t + \Delta t$. The operator prioritizes the removal of demand nodes with higher path blockage risk indices.

During the repair phase, the predictive collaborative insertion operator dynamically selects the service mode by jointly considering predicted road conditions and drone flight risks. When road blockage is imminent, but drone flight conditions remain safe, drones are prioritized to avoid vehicles becoming stranded en route. When road blockage is imminent and drone flight risks are also elevated, vehicles should be dispatched promptly to complete delivery before the road becomes impassable.

The pseudocode for the EALNS algorithm is presented in Table 3.

Table 3. The pseudocode for the EALNS algorithm.

Algorithm 1 EALNS

Input: Initial feasible solution S_0, cooling rate ε, initial temperature T_0, maximum iterations maxIter
  $S^* \leftarrow S_0$, $S \leftarrow S_0$ Initialise current and global best solutions
  Initialise weights for all destroy and repair operators: ω_d, ω_r = 1
  Initialise operator scores π = 0 and invocation counts θ = 0
  $T \leftarrow T_0$ Initialise temperature
  iter←0 Initialise iteration counter
while iter < maxIter **do**
    Select the destroy operator *d* and repair operator *r* via roulette wheel selection
    $S_t \leftarrow d(S)$ Destroy current solution
    $S' \leftarrow r(S_t)$ Repair to generate new solution
    Apply 2-opt local search to S.
    if $obj(S') < obj(S^*)$ **then**
        $S^* \leftarrow S'$
        $S \leftarrow S'$
        $\pi_d \leftarrow \pi_d + \sigma_1$, $\pi_r \leftarrow \pi_r + \sigma_1$ Update operator scores
    else if $obj(S') < obj(S)$ **then`**
        $S \leftarrow S'$
        $\pi_d \leftarrow \pi_d + \sigma_2$, $\pi_r \leftarrow \pi_r + \sigma_2$
    else
        Compute acceptance probability $p = exp(\,(obj(S) - obj(S'))\,/\,T\,)$
        if rand(0,1) < p **then**
            $S \leftarrow S'$
            $\pi_d \leftarrow \pi_d + \sigma_3$, $\pi_r \leftarrow \pi_r + \sigma_3$
        end if
    end if
  Update temperature $T \leftarrow T \times \varepsilon$
  Update invocation counts $\theta_d \leftarrow \theta_d$+1, $\theta_r \leftarrow \theta_r$+1
  if segment length is reached **then**
    for each operator i **do**
      Update weights $\omega_i \leftarrow \lambda \cdot \omega_i + (1 - \lambda) \cdot (\pi_i / \theta_i)$
    end for
    Reset operator scores and invocation counts $\pi \leftarrow 0$, $\theta \leftarrow 0$
  end if
  iter←iter+1
end while
Output S^*

5. Numerical experiment

To validate the effectiveness of the collaborative route optimization model for emergency vehicles and drones under dynamically evolving heavy rainfall-induced urban flooding, as well as the EALNS algorithm, systematic numerical experiments are conducted using Zhongbei Town, Xiqing District, Tianjin as the study area, simulating emergency supply distribution scenarios during flood conditions.

The experiments comprise three principal components, each designed to validate the model and algorithm from different perspectives. First, the EALNS algorithm is benchmarked against the commercial solver Gurobi to verify the correctness of the mathematical model and evaluate the baseline performance of EALNS in terms of computational efficiency and solution accuracy. Second, a comparative analysis is conducted between ALNS, GA, and EALNS, with particular emphasis on assessing the improvements in solution quality, convergence speed, and robustness. Finally, a comparison between the vehicle–drone collaborative mode and the traditional vehicle-only mode is performed to quantitatively demonstrate the practical value of the proposed collaborative mechanism and the strategy accounting for dynamic flood evolution.

All experiments are conducted over multiple independent runs, with performance metrics, including mean values and standard deviations, recorded to ensure the reliability and stability of the experimental findings.

5.1. Data preparation

Based on actual road network data from Zhongbei Town, Xiqing District, Tianjin, we systematically collect and process the regional road topology. The urban road network is abstracted into a directed graph model to represent spatial structural relationships for emergency rescue route planning, as illustrated in Figure 3. Within this directed graph, node J51 corresponds to the depot location, serving as the starting point for emergency rescue vehicles and drones. Nodes J132, J177, and J193 correspond to distribution centers.

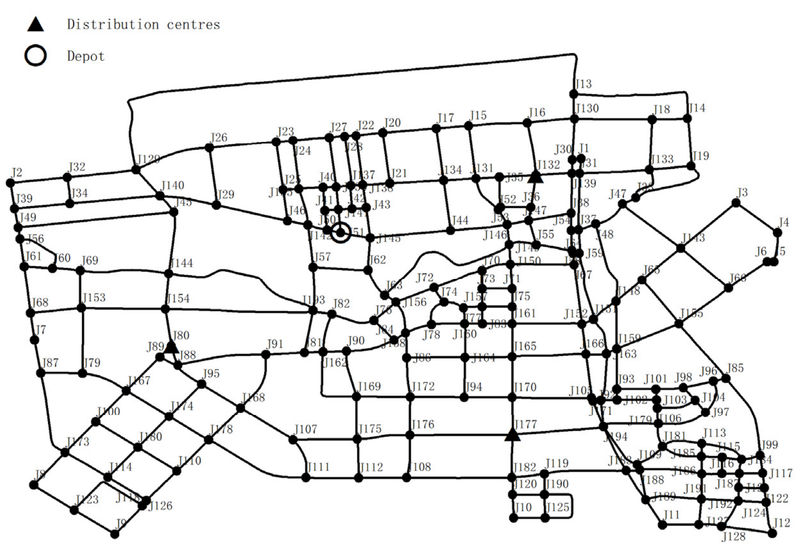


Figure 3. The urban road network of the study area.

The design storm pattern is established based on the rainfall intensity formula and design hyetograph standards specified in the Code for Design of Outdoor Wastewater Engineering (GB 50014–2006). The rainfall intensity is expressed as:

$$i = \frac{q}{167} = \frac{A \cdot \left(1 + C \times \lg T\right)}{167 \times (t+b)^{n}} \tag{45}$$

In Equation (45), i denotes rainfall intensity (mm·min^{-1}); t represents duration (min); T denotes the return period (years); and A, b, C, and n are empirical parameters determined using the Levenberg–Marquardt algorithm, yielding A = 2141, b = 9.6093, C = 0.7562, and n = 0.6893, respectively.

We specify a storm duration of 120 minutes and a design return period of 50 years. By substituting the relevant parameters into the rainfall intensity formula, the design storm hyetograph is derived. This hyetograph is subsequently input into the rain gauge module of the Storm Water Management Model (SWMM) to generate a rainfall time series conforming to design specifications, thereby completing the configuration of the storm rainfall pattern.

Based on the actual characteristics of the study area and recommendations from the SWMM User's Manual, key model parameters are, established as detailed in Table 4. The SWMM model is then employed to simulate the dynamic variation of inundation depth across the urban road network under the storm scenario.

Table 4. SWMM model parameters.

Model Parameter	Value
Manning’s roughness coefficient for pervious areas	0.21
Maximum depression storage for pervious areas (m)	0.0069
Manning's roughness coefficient for impervious areas	0.013
Maximum depression storage for impervious areas (m)	0.0023
Drainage pipe diameter (m)	2
Pipe roughness coefficient	0.014

Within the study area, statistical analysis and compilation of length and width data for all relevant road sections are conducted. Based on the overflow data output from the SWMM model and Manning's formula, water depth levels for each section under waterlogging conditions are subsequently calculated. The design travel speed v_0 for sections without waterlogging is determined according to the urban road design speed table, as shown in Table 5. Considering the vehicle-type characteristics of the selected emergency response vehicles, and following the conservative assumptions adopted by Tsang and Scott [63] in modeling flood-induced emergency service accessibility, as well as the mechanistic insights on vehicle instability in floodwaters reported by Al-Qadami et al. [64], the critical waterlogging depth d_p permitting emergency vehicle passage is set at 30 cm. Road segments with water depths exceeding d_p are considered impassable for emergency vehicles in the model.

Table 5. Urban road design speeds.

Road Classification	Expressway	Arterial road	Collector road	Local road
Design speed (km/h)	80	60	40	30

By substituting the time-series road waterlogging depth data into Equation (4), the travel speed of emergency vehicles at the mth simulation output time step can be obtained. To accurately characterize the travel speed of emergency vehicles along road sections, the average of output speeds from two adjacent simulation time steps is calculated and adopted as the representative speed for emergency vehicle travel during that interval. Consequently, a dynamic matrix of vehicle speed variations is derived. Based on this speed variation information and incorporating the lengths of the relevant road segments, the dynamic travel cost matrix presented in Table 6 is constructed.

Table 6. Partial matrix of emergency vehicle travel time.

Adjacent node	Emergency vehicle travel time by output time step/s				
number	(35,36] min	(36,37] min	(37,38] min	(38,39] min	(39,40] min
$J_{13}J_{14}$	155.64	156.67	158.12	160.24	163.44
$J_{13}J_{130}$	33.41	33.52	33.67	33.88	34.2
$J_{15}J_{16}$	70.11	73.37	78.25	85.78	97.87
...	...	...	...	...	...
$J_{15}J_{131}$	48.66	48.91	49.24	49.71	50.4
$J_{16}J_{130}$	55.65	58.07	61.69	67.28	76.25
$J_{18}J_{133}$	71.8	73.2	75.21	78.19	82.78

Despite the advantages of drones, such as high flight speed and independence from ground traffic constraints, their flight performance may be adversely affected by rainstorm conditions, thereby reducing the efficiency of relief supply delivery. Therefore, in this section, we systematically configure the weather time window classification, wind velocity and direction parameters, and drone flight risk probabilities based on the actual meteorological characteristics of the study area.

Table 7. Drone flight risk probability settings.

Wind Scale	Wind Speed Range (m/s)	Risk Probability	Short-duration Rainfall Intensity	Rainfall (mm/h)	Risk Probability
⩽2	⩽3.3	0%	Light	30.0~49.9	10%
3	3.4~5.4	5%			
4	5.5~7.9	10%	Moderate	50.0~69.9	40%
5	8.0~10.7	20%			
6	10.8~13.8	40%	Heavy	70.0~99.9	70%
7	13.9~17.1	60%			
8	17.2~20.7	80%	Extreme	⩾100	100%
⩾9	⩾20.8	100%			

Wind velocity and direction are critical meteorological factors affecting drone flight efficiency and safety. When the drone's airspeed remains constant, wind velocity and direction influence its ground speed, which in turn affects the actual flight time and operational risk. Rainfall intensity is another key factor affecting drone flight safety under rainstorm conditions. With reference to wind and rainfall classification standards and the weather-resistance characteristics of the selected DJI drone, the drone risk probability settings are configured, as shown in Table 7. The maximum tolerable risk threshold for drones is set at 40%.

Table 8. Partial matrix of risk probability.

Time Period	Wind Direction (°)	Wind Speed (m/s)	$\rho_w^{(m)}$	$\rho_r^{(m)}$	$\rho^{(m)}$
0:01	247.17	3.09	0%	0%	0%
0:02	247.17	3.09	0%	0%	0%
0:03	247.17	3.09	0%	0%	0%
…	…	…	…	…	…
0:59	247.17	3.09	0%	70%	70%
1:00	277.33	3.53	5%	70%	71.5%
…	…	…	…	…	…
1:58	277.33	3.53	5%	0%	5%
1:59	277.33	3.53	5%	0%	5%
2:00	277.33	3.53	5%	0%	5%

Based on historical meteorological monitoring data from the study area, we obtain the wind speed $v_w^{(m)}$,wind direction θ_m, and rainfall intensity for each weather time window, which are then used to construct the weather scenarios for the case analysis. Table 8 presents an example of the processed weather data. The composite risk probability $\rho^{(m)}$ is calculated using Equation (46), where $\rho_w^{(m)}$ and $\rho_r^{(m)}$ denote the risk probabilities associated with wind and rainfall, respectively.

$$\rho^{(m)} = 1 - (1 - \rho_w^{(m)}) \cdot (1 - \rho_r^{(m)}) \tag{46}$$

5.2. Problem instances and parameter settings

To ensure scientific rigor and reproducibility, we provide a systematic description of the experimental environment, problem scale, key parameters, and the rationale for their configuration as adopted in the subsequent numerical experiments.

To comprehensively validate the effectiveness of the proposed model and algorithm, two categories of test cases are designed: benchmark instance testing and real-world case analysis. Due to the absence of a standardized benchmark dataset for this problem, the classical Solomon VRPTW benchmark instances are adopted as the basis, with drone-related collaborative delivery parameters incorporated to construct simulation instances appropriate for this study. For the case analysis, the road network of Zhongbei Town, Xiqing District, Tianjin, isselected as the study area. The road network is abstracted as a directed graph comprising 194 demand nodes and 3 distribution center nodes. The key parameter settings are specified as follows: The waterlogging-related parameters, including the normal road design speed v_0, the time-dependent road water depth $x_{ij}^{(m)}$, and the maximum traversable waterlogging depth

d_p, are obtained or specified according to the procedures described in Section 5.1. Based on the calculations presented in the preceding section, the maximum capacity of emergency vehicles and drones, Q_k and W_u, are set to 50,000 and 2,500 units, respectively. Following Agatz et al. [65], the drone airspeed is set to 1 km/min, and the maximum endurance E_u is set to 30 minutes. The cost parameters are assigned with reference to the cost settings in Zhen et al. [66] and Gao et al. [67]. Accordingly, the response cost objective incorporates distance-based transportation costs and fixed deployment costs, where c^k and c^u denote the unit transportation costs of vehicles and drones, respectively, and c_f^k and c_f^u denote their corresponding fixed deployment costs. The parameter configuration for the EALNS algorithm is as follows: weight update factor λ = 0.65; operator reward scores σ_1 = 30, σ_2 = 15, and σ_3 = 5; maximum number of iterations Maxiter = 500; simulated annealing initial temperature T = 150; and cooling rate rate = 0.995. The algorithm is implemented in Python 3.9 and executed on a computer equipped with an Intel Core i7-14650HX processor (2.20 GHz) and 32 GB of RAM, running Windows 11.

5.3. Model and algorithm validation

To validate the correctness of the mathematical model proposed in this study and the effectiveness and computational efficiency of the EALNS algorithm, comparative experiments are conducted on benchmark instances of different scales in this subsection. For each instance, two additional distribution centers are randomly introduced into the original benchmark to generate the corresponding test data. The Gurobi solver is employed to solve various test instances based on the proposed model, with results compared against those obtained by the EALNS algorithm, as detailed in Table 9, where K and D denote the number of emergency vehicles and drones deployed, respectively. Moreover, we utilize the classical Solomon-VRPTW benchmark dataset, incorporating parameters pertinent to drone-assisted collaborative delivery to construct simulation instances suitable for this research. The test instances encompass three demand point distribution types, namely C-type clustered distribution, R-type random distribution, and RC-type random-clustered hybrid distribution, as well as two time window configurations, namely wide time windows and narrow time windows.

The maximum runtime for the Gurobi solver is set to 1800 seconds. The performance gap is calculated as $Gap = (TC(EALNS) - TC(Gurobi)) \,/\, TC(Gurobi) \times 100$, where a negative value indicates that the EALNS algorithm outperforms Gurobi.

As shown in Table 9, EALNS obtains solutions that are equal to or better than the Gurobi incumbent in most small- and medium-scale instances, while showing only minor deviations in a few cases. The maximum positive gap is only 0.42%, indicating that EALNS maintains competitive solution quality even when it does not obtain the best result. As the number of demand nodes increases, Gurobi fails to converge to an optimal solution within the 1,800-second time limit in most instances, reflecting the inherent computational limitations of MIP formulations; the exponential time complexity renders exact solving increasingly intractable for medium- and large-scale problems. By contrast, the EALNS algorithm consistently delivers high-quality feasible solutions with shorter average runtimes across all test instances. Notably, for instances c102-20, c103-50, and r101-50, the solutions produced by EALNS surpass the best incumbents obtained by Gurobi within the time limit. For the medium-scale instances with 60 and 75 demand nodes, the computational burden of Gurobi increases significantly, whereas EALNS obtains high-quality feasible solutions within much shorter computational times. In terms of computational efficiency, the EALNS algorithm holds a substantial advantage: As problem scale grows, the runtime of exact algorithms increases exponentially, whereas

EALNS mitigates premature convergence to local optima by accepting inferior solutions with a controlled probability, thereby sustaining broader exploration of the solution space and consistently producing higher-quality feasible solutions.

Table 9. Comparison of results between Gurobi and EALNS on small and medium instances.

Instance	K	D	Gurobi		EALNS		Gap (%)
			TC	RT(s)	TC	RT(s)	
c101-10	1	2	864.41	1793.33	864.41	1.59	0.00
c102-20	2	4	1940.95	1800	1923.47	3.46	−0.90
c103-50	3	6	5047.09	1800	5040.79	19.16	−0.12
c201-10	1	1	951.45	121.34	951.45	1.78	0.00
c202-20	2	4	1990.23	1800	1998.51	3.37	0.42
c203-50	3	6	5108.08	1800	5108.08	17.43	0.00
r101-10	1	1	268.96	1.09	268.964	1.65	0.00
r101-20	2	4	499.47	1800	499.47	4.22	0.00
r101-50	3	6	1226.21	1800	1223.53	41.29	−0.22
r109-10	1	1	268.96	0.78	268.96	1.60	0.00
r109-20	2	4	535.00	1800	535.00	4.52	0.00
r109-50	3	6	1224.52	1800	1202.21	26.83	−1.82
r202-10	1	1	278.25	1.61	278.25	1.34	0.00
r202-20	2	4	499.47	1800	499.47	4.28	0.00
r203-50	3	6	1239.54	1800	1205.34	21.60	−2.75
rc101-10	1	2	235.94	1245.27	235.94	1.27	0.00
rc102-20	2	4	524.79	1800	525.77	10.51	0.19
rc203-50	3	6	1351.01	1800	1334.68	17.97	−1.21
c105-60	3	6	6124.33	1800	6128.95	24.09	0.08
c205-60	3	6	6240.58	1800	6240.58	25.73	0.00
r102-60	3	6	1396.30	1800	1383.09	34.40	−0.95
r201-60	3	6	1408.06	1800	1389.41	34.54	−1.37
rc103-60	3	6	1764.24	1800	1746.53	27.26	−1.00
c105-75	4	8	7570.91	1800	7513.08	64.41	−0.76
c205-75	4	8	7612.36	1800	7593.18	64.25	−0.25
r105-75	4	8	1553.12	1800	1550.46	78.46	−0.17
r201-75	4	8	1576.09	1800	1551.39	81.59	−1.57
rc204-75	4	8	1783.92	1800	1761.31	61.52	−1.27

Table 10. Comparison of EALNS with ALNS, GA, and WOA on large-scale instances.

Instance	EALNS	ALNS	GA	WOA
c104-100	9903.33	9959.97	9899.71	9958.12
c201-100	10073.72	10127.23	10152.26	10206.51
r102-100	1916.74	1952.31	1971.28	1986.82
r209-100	1897.26	1929.84	1941.76	1965.32
rc103-100	2237.12	2274.68	2301.42	2289.35

The results in Table 10 further verify the performance of EALNS on large-scale instances. Overall, EALNS obtains lower objective values than the compared algorithms in most cases, demonstrating the effectiveness of the proposed enhancement mechanisms. In c104-100, GA obtains a slightly better result than EALNS, but the difference is only approximately 0.04%, indicating that EALNS still achieves a highly competitive solution. Overall, EALNS outperforms ALNS, GA, and WOA in most large-scale instances, which demonstrates that the proposed EALNS algorithm can achieve competitive solution quality, thereby validating the effectiveness of the algorithm design. Combined with the medium-scale results in Table 9, these findings further confirm the scalability and robustness of EALNS under different problem sizes and distribution patterns.

Before conducting the comparative algorithmic evaluation, the ε-constraint method is first used to verify the generation of nondominated solutions for the Tianjin Xiqing case under the proposed multi-objective framework.

Table 11. Partial nondominated solutions obtained by the ε-constraint method.

Solution	Total response time f_1	Total response cost f_2	Average demand penalty f_3
1	8237.78	1254.96	0.54
2	7576.36	1305.79	0.48
3	7347.4	1504.93	0.36
4	7192.53	1505.42	0.27

As shown in Table 11, increasing the total response cost leads to reductions in total response time and average demand-point penalty, indicating that additional resource deployment can improve response efficiency and service timeliness. Based on these trade-offs, a compromise solution is selected from the nondominated solution set using the ideal-point-based method.

Following the comparison with Gurobi, we further investigate the overall performance of the EALNS algorithm in real-world urban environments and dynamic flooding scenarios by benchmarking it against three representative metaheuristic algorithms: the standard Adaptive Large Neighborhood Search (ALNS), the Genetic Algorithm (GA), and the Whale Optimization Algorithm (WOA). To ensure comparability, all algorithms are implemented using the same case data, objective function, constraint set, maximum number of iterations, and computational environment. For GA, the parameter settings follow Liu et al. [68], with the population size set to 100, crossover rate to 0.8, and mutation probability to 0.1. A route-based chromosome is adopted to encode the routing decisions. Fitness-based selection and route-exchange crossover are used to generate offspring, while route-level mutation is performed through node swapping, subsequence reversal, and node relocation. The WOA benchmark is implemented based on the enhanced WOA of Wang et al. [69]. Since the vehicle–drone collaborative routing problem is a discrete combinatorial optimization problem, each whale is encoded as a complete routing solution. The encircling, global search, and spiral updating behaviors of WOA are implemented through route-segment inheritance from the current best or elite solutions and route-level perturbations, including shuffling, swapping, reversing, and relocation. ALNS adopts the conventional adaptive large neighborhood search framework, and its major parameters are set with reference to the recommended values in Keskin and Çatay [70]. In each iteration, a destroy operator and a repair operator are selected according to their adaptive weights. The destroy operators include random removal, Shaw removal, worst-cost removal, and route removal, whereas the repair operators include greedy insertion and regret-2 insertion. Zhongbei Town in Xiqing District, Tianjin is chosen as the study area.

To comprehensively evaluate algorithmic performance, each algorithm is independently executed ten times, with the following metrics recorded: Optimal objective function value (Obj. Best), reflecting the algorithm's peak optimization capability; average objective function value (Obj. Avg); and standard deviation (Obj. Std), jointly characterizing the algorithm's robustness and stability. The comparative results are presented in Table 12.

Table 12. Performance evaluation of algorithms for the dynamic collaborative vehicle routing problem.

Metric	GA	ALNS	EALNS	WOA
Obj. Best	9780.81	6035.88	5379.44	8610.81
Obj. Avg	10728.02	7623.57	6182.41	9419.36
Obj. Std	684.35	647.12	416.49	749.38

Based on the results presented in Table 12, EALNS algorithm's performance can be analyzed as follows: In terms of solution quality, the optimal and average objective values obtained by EALNS surpass those of ALNS and GA. Specifically, EALNS achieves an optimal objective value of 5379.44, representing improvements of approximately 45%, 10.9%, and 37.5% compared to GA (9780.81), ALNS (6035.88), and WOA (8610.81), respectively. These results demonstrate that the EALNS algorithm is more effective in exploring the complex solution space under dynamic rainstorm and waterlogging conditions. In terms of robustness, EALNS exhibits strong stability, with a standard deviation of 416.49 across ten independent runs. This consistency is crucial for ensuring reliability in practical applications.

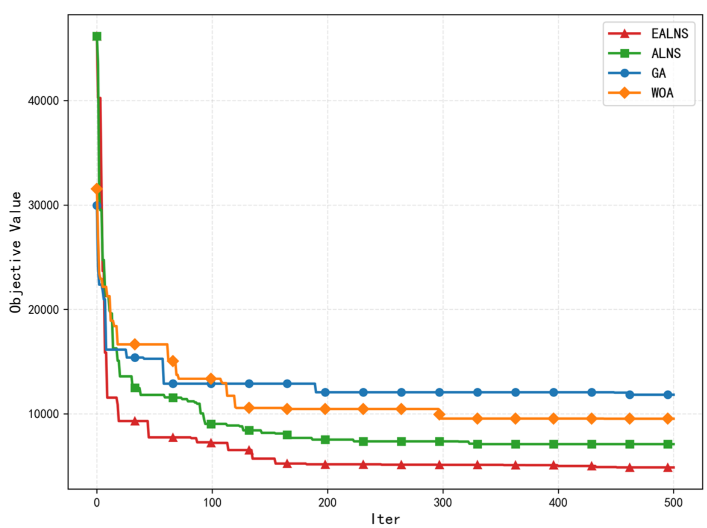


Figure 4. Convergence curves.

Figure 4 illustrates the convergence curves of EALNS, ALNS, GA, and WOA during the solution process. The four algorithms exhibit distinct convergence behaviors. Compared to GA, ALNS, and WOA, EALNS demonstrates the fastest convergence rate, with its objective function value decreasing rapidly during the initial iterations and stabilizing at a superior level. In contrast, ALNS, GA, and WOA exhibit relatively slower initial convergence, requiring more iterations to reach stable states. The ALNS curve shows diminishing improvement after approximately 100 iterations, whereas EALNS continues to improve. This suggests that the enhancements incorporated in EALNS, including the simulated annealing acceptance criterion, problem-oriented operator design, and adaptive operator selection, contribute to enhanced global exploration capabilities, thereby yielding higher-quality solutions.

In summary, the experimental results demonstrate that EALNS outperforms ALNS, GA, and WOA in convergence speed and solution quality under the dynamic rainstorm and waterlogging delivery scenario considered in this study. These findings validate the effectiveness and practicality of the proposed algorithm for solving the drone-vehicle collaborative routing problem in complex real-world environments.

5.4. Value Analysis of the Collaborative Model

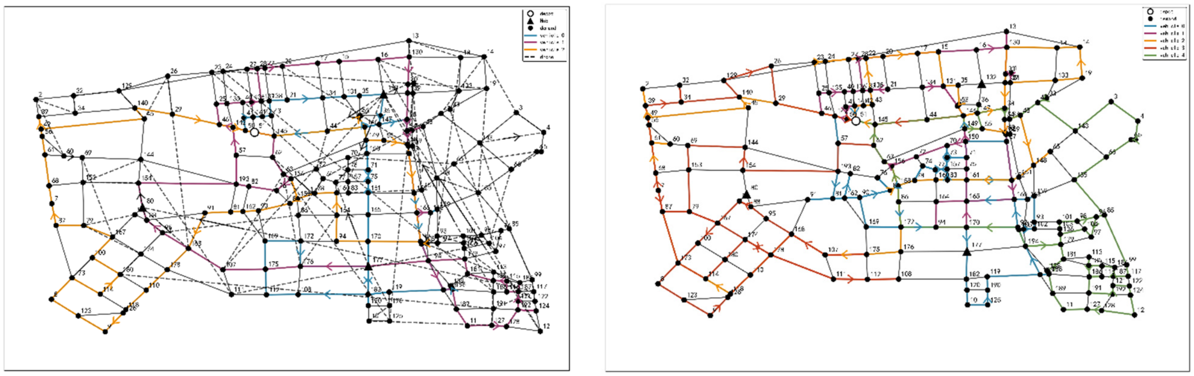
To evaluate the practical value of the vehicle-drone collaborative rescue model under dynamically evolving heavy rainfall-induced urban flooding conditions, we compare it with the vehicle-only mode based on four metrics: total rescue time, total rescue cost, average penalty, and demand point service rate. The comparison results are presented in Table 13 and Figure 5.

Table 13. Comparison between the vehicle-only and the vehicle-drone collaborative models.

Metric	Vehicle-only	Vehicle-drone collaborative
Total time	7106.60	6308.74
Total cost	1000.00	900.00
Avg. penalty	0.2944	0.1783
Service rate	100.00%	100.00%

As shown in Table 13, the vehicle–drone collaborative model outperforms the vehicle-only mode across all rescue metrics. In terms of total rescue time, the collaborative model achieves 6308.74 seconds, representing an 11.2% reduction compared with the vehicle-only mode. Regarding total rescue cost, the collaborative model incurs 900.00 units, a 10.0% decrease relative to the vehicle-only mode. This improvement is primarily attributable to drones undertaking low-load, high-urgency delivery tasks, thereby reducing the total travel distance and time cost of emergency vehicles. For the average penalty, the collaborative model achieves 0.1783, representing a 39.4% reduction compared with the vehicle-only mode, indicating that rescue timeliness requirements under time window constraints are satisfactorily fulfilled. Regarding the demand point service rate, both modes maintain full-coverage rescue capability while further validating resource scheduling efficiency. Detailed routing results are presented in Figure 5.

The core advantage of the vehicle-drone collaborative model stems from its resource complementarity within dynamic environments. On the one hand, drones possess high maneuverability and spatial traversal capabilities, enabling rapid delivery of lightweight, urgent supplies to areas inaccessible to vehicles, thereby avoiding time losses caused by route obstructions or delays. On the other hand, the collaborative mechanism achieves efficient division of labor through a distribution center relay design, whereby vehicles handle trunk-line transport while drones manage last-mile delivery, fully exploiting the flexibility and speed advantages of unmanned aerial vehicles.



(a) (b)

Figure 5. Detailed routing results.

In summary, the vehicle–drone collaborative model demonstrates adaptability under dynamically evolving heavy rainfall-induced urban flooding conditions. Compared with the vehicle-only mode, it offers advantages in response time, cost, and penalty reduction, thereby enhancing the resilience and adaptability of emergency rescue operations.

5.5. Sensitivity analysis

Several scholars have investigated the parameters of vehicles and drones. Lehmann and Winkenbach [71] conducted sensitivity analyses on vehicle capacity and range, while Liu et al. [72] analyzed drone design parameters. To investigate the sensitivity of relief supply transport to carrier capacity, we perform sensitivity analyses on vehicle and drone capacity parameters, examining their effects on the system objective function. Identifying optimal capacity configurations for vehicles and drones facilitates the development of more efficient transport systems under constrained conditions.

Figure 6 presents the results of the sensitivity analysis for capacity parameters. The experimental results indicate that the objective function value decreases progressively as capacity increases. Specifically, when vehicle capacity increases from 30,000 to 60,000 units, the objective function value declines, with marginal improvement diminishing in the range of 45,000 to 60,000 units as the curve flattens. This suggests that the emergency vehicle capacity is approaching sufficiency, and further increases would yield negligible benefits. Drone capacity exhibits a similar pattern: The objective function value decreases continuously within the range of 3,000 to 9,000 units, with diminishing marginal returns observed between 8,000 and 9,000 units. A comparison of the effects of drone and vehicle capacity reveals that changes in drone capacity exert a greater influence on the results. Therefore, if funding is available, prioritizing the expansion of drone capacity is recommended.

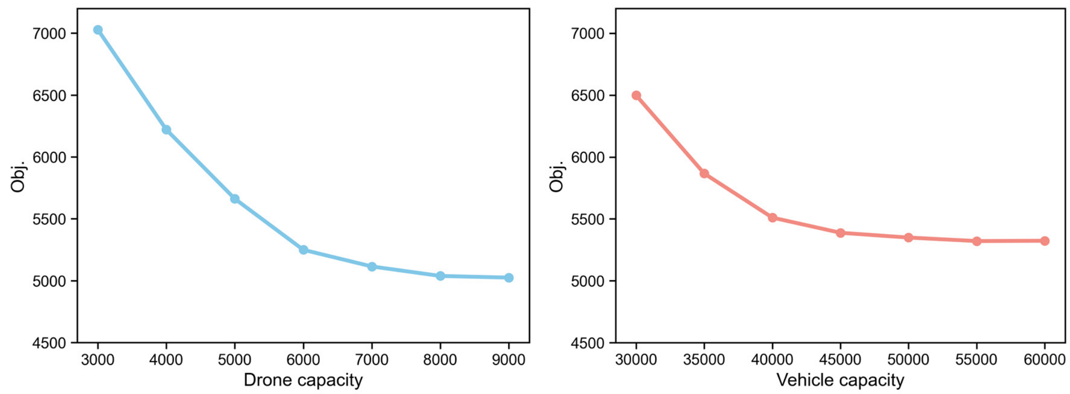


Figure 6. Sensitivity analysis of capacity parameters.

As capacity increases, the optimization algorithm can more flexibly consolidate neighboring demand nodes, reducing unnecessary detours and redundant trips, thereby lowering variable costs and total rescue time. Furthermore, improved time window compliance contributes significantly to the improvement in the objective function. Sufficient carrying capacity enables a single trip to serve more urgent demands, reducing the risk of service delays caused by multiple round trips and effectively minimizing time window violation penalties. However, once capacity exceeds a certain threshold, further increases yield saturating returns.

6. Conclusions

In this paper, we investigate the collaborative route optimization problem for urban emergency rescue vehicles and drones under dynamically evolving, heavy rainfall-induced urban flooding conditions. We comprehensively consider key factors, including reduced road trafficability due to waterlogging, rescue timeliness requirements, and time window constraints. A multi-objective mixed-integer programming model is formulated to minimize the total rescue time, total cost, and average penalty. The model incorporates a dynamic vehicle speed function influenced by waterlogging depth, a time-varying drone ground speed model accounting for wind velocity and direction, and a soft time window penalty mechanism. These components effectively capture the temporal variations in vehicle traversability and drone flight efficiency as urban flooding evolves, while ensuring drone flight safety through cumulative risk threshold constraints. To address the NP-hard nature of this problem, an enhanced adaptive large neighborhood search algorithm (EALNS) is developed, integrating dynamic operator selection, problem-oriented operator design, simulated annealing acceptance criteria, predictive search mechanisms, and local search enhancement strategies. Experiments conducted using real-world road network data and simulated heavy rainfall-induced flooding data from Zhongbei Town, Xiqing District, Tianjin demonstrate that the proposed method exhibits superior performance in computational efficiency and solution quality. Furthermore, comparative analysis reveals that the vehicle–drone collaborative mode achieves significant improvements in rescue time, cost, and timeliness compared with traditional vehicle-only modes. Sensitivity analysis reveals that the objective function value decreases monotonically with increasing capacity, exhibiting asymptotic convergence. This behavior reflects the combined influence of economies of scale and the law of diminishing marginal returns. Accordingly, the optimal capacity allocation range should be determined through numerical analysis to maximize rescue efficiency by fully exploiting scale advantages while avoiding resource redundancy. Based on these findings, the following managerial insights and practical recommendations are proposed:

(1) In urban flood disaster emergency response, vehicle–drone collaborative rescue should be prioritized over conventional vehicle-only approaches. Comparative analysis demonstrates that the collaborative mode reduces total rescue time by 11.2%, total cost by 10.0%, and average time-window penalty by 39.4%. When floodwater depth on road segments exceeds the passability threshold, the EALNS algorithm is triggered to automatically re-plan routes, enabling dynamic adjustment. Emergency management authorities should therefore integrate drone fleets into rescue systems; for demand points where severe road flooding renders vehicle access infeasible, drones can enable precise supply delivery, effectively improving rescue coverage and response timeliness.

(2) Numerical analysis should be conducted for specific scenarios to optimally configure the transport capacities of vehicles and drones. Sensitivity analysis reveals that the objective function value decreases monotonically with increasing transport capacity, with marginal benefits gradually approaching saturation beyond a certain threshold. Furthermore, variations in drone capacity exert a greater influence on overall system performance than those in vehicle capacity. Accordingly, management authorities should conduct targeted numerical experiments based on local disaster characteristics and resource endowments to fully exploit economies of scale while avoiding resource redundancy. Under resource-constrained conditions, prioritizing the expansion of drone transport capacity is recommended.

(3) The differential impacts of dynamic environmental factors on vehicles and drones warrant careful consideration. This study demonstrates that road waterlogging depth significantly affects emergency vehicle travel speeds, while wind velocity and direction directly influence drone ground speeds and flight safety. Accordingly, when formulating rescue plans, emergency management authorities should establish environmental monitoring and early warning mechanisms. By comprehensively assessing waterlogging conditions and meteorological factors, vehicle routing and drone deployment strategies can be dynamically adjusted, thereby enhancing the resilience and adaptability of rescue systems in dynamic environments.

(4) Government emergency management authorities should develop a formal mechanism for the emergency requisition of drones, enabling the rapid mobilization of operational teams from large-scale drone operators, including courier and logistics companies, during disaster response. This mechanism should include clearly defined compensation standards and a centralized resource registry database to maximize the operational value of requisitioned assets in rescue missions.

Building on the managerial insights outlined above, these findings highlight the practical importance of integrating drones with traditional vehicle-based rescue operations. This combination not only enhances efficiency and reduces costs but also provides flexibility to reach areas inaccessible by road, ensuring timely delivery of critical supplies. Optimizing transport capacity, particularly by prioritizing drone deployment, can further improve overall system performance while avoiding resource redundancy. Dynamic environmental conditions, such as road waterlogging and wind variations, underscore the need for real-time monitoring and adaptive routing, enabling managers to respond proactively to changing conditions. Additionally, formal mechanisms for rapid mobilization of external drone resources, including partnerships with logistics providers and clear operational procedures, can help strengthen emergency response. Together, these considerations demonstrate how combining technological innovation, adaptive resource allocation, and flexible operational planning can effectively improve the efficiency and resilience of urban flood rescue systems, offering actionable guidance for emergency managers.

Despite these contributions, several limitations remain that warrant further investigation. First, the model restricts each demand point to service by a single transport mode. However, in large-scale disasters, relief requirements at certain critical locations may exceed the capacity of any individual vehicle or drone. In future studies, researchers could relax this constraint to permit joint service by multiple transport modes, necessitating the development of synchronization mechanisms to coordinate arrival times and resource handovers. Second, we assume that the locations and demands of affected areas are known with certainty at the time of decision-making. In practice, disaster scenarios are characterized by substantial uncertainty: The number of affected individuals, the severity of damage, and the quantity of required supplies often cannot be precisely estimated in the immediate aftermath of an event. Thus, researchers could incorporate stochastic programming or robust optimization approaches to address demand uncertainty, thereby generating solutions that remain effective across a range of plausible scenarios. Third, the proposed model treats vehicles and drones as homogeneous fleets with identical specifications. Extending the framework to accommodate heterogeneous fleets, including vehicles and drones with varying capacities, speeds, and operational ranges, would better reflect the equipment configurations available to emergency management agencies. Finally, we assume homogeneous cargo with uniform handling requirements. In actual rescue operations, different categories of materials possess distinct urgency levels and storage constraints. Incorporating relief supplies diversity and prioritization would enhance the practical applicability of the model.

Author contributions

Yuanbo Zhang: Conceptualization, methodology, software, validation, formal analysis, investigation, writing-original draft, visualization; Na Li: Conceptualization, resources, supervision, project administration, funding acquisition, writing-review & editing.

Use of Generative-AI tools declaration

Generative AI tools were used solely to assist in language refinement, grammar checking, and improving clarity of expression during the manuscript preparation. All conceptual development, framework design, empirical analysis, and interpretation of results were conducted entirely by the authors. No AI tools were used for data analysis, methodological design, or generation of research findings.

Acknowledgments

No funding was received.

Conflict of interest

All authors declare no conflicts of interest in this paper.

References

1. J. Chen, X. Shi, L. Gu, G. Wu, T. Su, H. M. Wang, et al., Impacts of climate warming on global floods and their implication to current flood defense standards, *J. Hydrol*., **618** (2023), 129236. https://doi.org/10.1016/j.jhydrol.2023.129236
2. B. F. Sanders, J. E. Schubert, D. T. Kahl, K. J. Mach, D. Brady, A. AghaKouchak, et al., Large and inequitable flood risks in Los Angeles, California, *Nat. Sustain*., **6** (2023), 47–57. https://doi.org/10.1038/s41893-022-00977-7
3. M. Yan, A. Wilson, J. L. Peel, S. Magzamen, Q. Sun, T. Li, et al., Community-wide Mortality Rates in Beijing, China, During the July 2012 Flood Compared with Unexposed Periods, *Epidemiology*., **31** (2020), 319–326. https://doi.org/10.1097/EDE.0000000000001182
4. J. C. L. Normand, E. Heggy, Assessing flash flood erosion following storm Daniel in Libya, *Nat. Commun*., **15** (2024), 6493. https://doi.org/10.1038/s41467-024-49699-8
5. K. T. C. Lim Kam Sian, P. Sagero, L. L. Kebacho, V. Ongoma, Dubai's record precipitation event of 16 April 2024–A diagnosis, *Atmos. Res*., **315** (2025), 107924. https://doi.org/10.1016/j.atmosres.2025.107924
6. Y. He, D. Ma, J. Xiong, W. Cheng, H. Jia, N. Wang, et al., Flash flood vulnerability assessment of roads in China based on support vector machine, *Geocarto Int*., **37** (2022), 6141–6164. https://doi.org/10.1080/10106049.2021.1926560
7. E. E. Koks, J. Rozenberg, C. Zorn, M. Tariverdi, M. Vousdoukas, S. A. Fraser, et al., A global multi-hazard risk analysis of road and railway infrastructure assets, *Nat. Commun*., **10** (2019), 2677. https://doi.org/10.1038/s41467-019-10442-3
8. P. Singh, V. S. P. Sinha, A. Vijhani, N. Pahuja, Vulnerability assessment of urban road network from urban flood, *Int. J. Disaster Risk Reduct*., **28** (2018), 237–250. https://doi.org/10.1016/j.ijdrr.2018.03.017
9. M. K. Banafaa, Ö. Pepeoğlu, I. Shayea, A. Alhammadi, Z. A. Shamsan, M. A. Razaz, et al., A Comprehensive Survey on 5G-and-Beyond Networks With UAVs: Applications, Emerging Technologies, Regulatory Aspects, Research Trends and Challenges, *IEEE Access*., **12** (2024), 7786–7826. https://doi.org/10.1109/ACCESS.2023.3349208
10. S. A. H. Mohsan, M. A. Khan, F. Noor, I. Ullah, M. H. Alsharif, Towards the Unmanned Aerial Vehicles (UAVs): A Comprehensive Review, *Drones*, **6** (2022), 147. https://doi.org/10.3390/drones6060147
11. T. I. Faiz, C. Vogiatzis, Md. Noor-E-Alam, Computational approaches for solving two-echelon vehicle and UAV routing problems for post-disaster humanitarian operations, *Expert Syst. Appl*., **237** (2024), 121473. https://doi.org/10.1016/j.eswa.2023.121473
12. M. Daneshvar, S. D. Jena, W. Rei, A two-stage stochastic post-disaster humanitarian supply chain network design problem, *Comput. Ind. Eng*., **183** (2023), 109459. https://doi.org/10.1016/j.cie.2023.109459
13. R. Van Steenbergen, M. Mes, W. Van Heeswijk, Reinforcement learning for humanitarian relief distribution with trucks and UAVs under travel time uncertainty, *Transp. Res. Part C Emerg. Technol*., **157** (2023), 104401. https://doi.org/10.1016/j.trc.2023.104401
14. X. Wang, T. M. Choi, H. Liu, X. Yue, A Novel Hybrid Ant Colony Optimization Algorithm for Emergency Transportation Problems During Post-Disaster Scenarios, *IEEE Trans. Syst. Man Cybern, Syst*., **48** (2018), 545–556. https://doi.org/10.1109/TSMC.2016.2606440

15. M. Lyu, Y. Zhao, C. Huang, H. Huang, Unmanned Aerial Vehicles for Search and Rescue: A Survey, *Remote Sens*., **15** (2023), 3266. https://doi.org/10.3390/rs15133266
16. M. Kucharczyk, C. H. Hugenholtz, Remote sensing of natural hazard-related disasters with small drones: Global trends, biases, and research opportunities, *Remote Sens. Environ*., **264** (2021), 112577. https://doi.org/10.1016/j.rse.2021.112577
17. G. Zhang, N. Jia, N. Zhu, Y. Adulyasak, S. Ma, Robust drone selective routing in humanitarian transportation network assessment, *Eur. J. Oper. Res*., **305** (2023), 400–428. https://doi.org/10.1016/j.ejor.2022.05.046
18. G. Macrina, L. Di Puglia Pugliese, F. Guerriero, G. Laporte, Drone-aided routing: A literature review, *Transp. Res. Part C Emerg. Technol*., **120** (2020), 102762. https://doi.org/10.1016/j.trc.2020.102762
19. L. Du, X. Li, Y. Gan, K. Leng, Optimal Model and Algorithm of Medical Materials Delivery Drone Routing Problem under Major Public Health Emergencies, *Sustainability*, **14** (2022), 4651. https://doi.org/10.3390/su14084651
20. Z. Ghelichi, M. Gentili, P. B. Mirchandani, Logistics for a fleet of drones for medical item delivery: A case study for Louisville, KY, *Comput. Oper. Res*., **135** (2021), 105443. https://doi.org/10.1016/j.cor.2021.105443
21. P. Kumar, K. Pal, M. C. Govil, Comprehensive Review of Path Planning Techniques for Unmanned Aerial Vehicles (UAVs), *ACM Comput. Surv*., **58** (2026), 1–44. https://doi.org/10.1145/3737280
22. C. C. Murray, A. G. Chu, The flying sidekick traveling salesman problem: Optimization of drone-assisted parcel delivery, *Transp. Res. Part C Emerg. Technol*., **54** (2015), 86–109. https://doi.org/10.1016/j.trc.2015.03.005
23. G. Wu, N. Mao, Q. Luo, B. Xu, J. Shi, P. N. Suganthan, Collaborative Truck-Drone Routing for Contactless Parcel Delivery During the Epidemic, *IEEE Trans. Intell. Transport. Syst*., **23** (2022), 25077–25091. https://doi.org/10.1109/TITS.2022.3181282
24. S. H. Huang, Y. H. Huang, C. A. Blazquez, C. Y. Chen, Solving the vehicle routing problem with drone for delivery services using an ant colony optimization algorithm, *Adv. Eng. Inform*., **51** (2022), 101536. https://doi.org/10.1016/j.aei.2022.101536
25. R. G. Ribeiro, L. P. Cota, T. A. M. Euzebio, J. A. Ramirez, F. G. Guimaraes, Unmanned-Aerial-Vehicle Routing Problem With Mobile Charging Stations for Assisting Search and Rescue Missions in Postdisaster Scenarios, *IEEE Trans. Syst. Man Cybern, Syst*., **52** (2022), 6682–6696. https://doi.org/10.1109/TSMC.2021.3088776
26. W. Peng, D. Wang, Y. Yin, T. C. E. Cheng, Multi-agent deep reinforcement learning-based truck-drone collaborative routing with dynamic emergency response, *Transp. Res. Part E Logist. Transp. Rev*., **195** (2025), 103974. https://doi.org/10.1016/j.tre.2025.103974
27. C. C. Murray, R. Raj, The multiple flying sidekicks traveling salesman problem: Parcel delivery with multiple drones, *Transp. Res. Part C Emerg. Technol*., **110** (2020), 368–398. https://doi.org/10.1016/j.trc.2019.11.003
28. M. R. Salama, S. Srinivas, Collaborative truck multi-drone routing and scheduling problem: Package delivery with flexible launch and recovery sites, *Transp. Res. Part E Logist. Transp. Rev*., **164** (2022), 102788. https://doi.org/10.1016/j.tre.2022.102788

29. T. Thomas, S. Srinivas, C. Rajendran, Collaborative truck multi-drone delivery system considering drone scheduling and en route operations, *Ann. Oper. Res*., **339** (2024), 693–739. https://doi.org/10.1007/s10479-023-05418-y
30. H. Zhou, H. Qin, C. Cheng, L. M. Rousseau, An exact algorithm for the two-echelon vehicle routing problem with drones, *Transp. Res. Part B Methodol*., **168** (2023), 124–150. https://doi.org/10.1016/j.trb.2023.01.002
31. O. Dukkanci, A. Koberstein, B. Y. Kara, Drones for relief logistics under uncertainty after an earthquake, *Eur. J. Oper. Res*., **310** (2023), 117–132. https://doi.org/10.1016/j.ejor.2023.02.038
32. Y. Gong, W. Wang, Y. Zhou, J. Cheng, Optimization of emergency material distribution routes in flood disaster with truck-speedboat-drone coordination, *J. Flood Risk Manag*., **18** (2025), e13045. https://doi.org/10.1111/jfr3.13045
33. R. Arumugaraj Nadar, J. K. Jha, J. J. Thakkar, Performance Evaluation of Emergency Medical Service Systems with Multiple Ambulance Types and Patient Types, *Int. J. Ind. Eng. Theory*., **31** (2024). https://doi.org/10.23055/IJIETAP.2024.31.4.9069
34. I. Acar, I. Altin, Multi-Depot General Colored Traveling Salesman Problem with Time Windows in Home Healthcare System: A Medication Delivery Example, *Int. J. Ind. Eng. Theory*., **32** (2025). https://doi.org/10.23055/IJIETAP.2025.32.2.10657
35. Z. Luo, M. Poon, Z. Zhang, Z. Liu, A. Lim, The Multi-visit Traveling Salesman Problem with Multi-Drones, *Transp. Res. Part C Emerg. Technol*., **128** (2021), 103172. https://doi.org/10.1016/j.trc.2021.103172
36. R. Gu, M. Poon, Z. Luo, Y. Liu, Z. Liu, A hierarchical solution evaluation method and a hybrid algorithm for the vehicle routing problem with drones and multiple visits, *Transp. Res. Part C Emerg. Technol*., **141** (2022), 103733. https://doi.org/10.1016/j.trc.2022.103733
37. S. Zhong, R. Cheng, Y. Jiang, Z. Wang, A. Larsen, O. A. Nielsen, Risk-averse optimization of disaster relief facility location and vehicle routing under stochastic demand, *Transp. Res. Part E Logist. Transp. Rev*., **141** (2020), 102015. https://doi.org/10.1016/j.tre.2020.102015
38. C. Cao, Y. Liu, O. Tang, X. Gao, A fuzzy bi-level optimization model for multi-period post-disaster relief distribution in sustainable humanitarian supply chains, *Int. J. Prod. Econ*., **235** (2021), 108081. https://doi.org/10.1016/j.ijpe.2021.108081
39. J. Zhao, Y. Guo, X. Duan, Dynamic Path Planning of Emergency Vehicles Based on Travel Time Prediction, *J. Adv. Transp*., **2017** (2017), 1–14. https://doi.org/10.1155/2017/9184891
40. J. F. Ehmke, A. M. Campbell, B. W. Thomas, Vehicle routing to minimize time-dependent emissions in urban areas, *Eur. J. Oper. Res*., **251** (2016), 478–494. https://doi.org/10.1016/j.ejor.2015.11.034
41. K. Liu, L. Yang, Y. Zhao, Z. H. Zhang, Multi-period stochastic programming for relief delivery considering evolving transportation network and temporary facility relocation/closure, *Transp. Res. Part E Logist. Transp. Rev*., **180** (2023), 103357. https://doi.org/10.1016/j.tre.2023.103357
42. F. Avishan, M. Elyasi, İ. Yanıkoğlu, A. Ekici, O. Ö. Özener, Humanitarian Relief Distribution Problem: An Adjustable Robust Optimization Approach, *Transp. Sci*., **57** (2023), 1096–1114. https://doi.org/10.1287/trsc.2023.1204

43. X. Yang, W. Cao, K. Wang, H. Yin, J. Wu, L. Wu, Integrated scheduling of truck and drone fleets for cargo transportation in post-disaster relief: A two-stage stochastic optimization approach, *Transp. Res. Part E Logist. Transp. Rev.*, **196** (2025), 104015. https://doi.org/10.1016/j.tre.2025.104015
44. K. Li, X. Yan, Y. Han, Multi-mechanism swarm optimization for multi-UAV task assignment and path planning in transmission line inspection under multi-wind field, *Appl. Soft Comput.*, **150** (2024), 111033. https://doi.org/10.1016/j.asoc.2023.111033
45. D. Chen, C. Imdahl, D. Lai, T. Van Woensel, A reinforcement learning approach for the dynamic vehicle routing and scheduling problem with stochastic request times and time-dependent, stochastic travel times, *Transp. Res. Part C Emerg. Technol.*, **182** (2026), 105387. https://doi.org/10.1016/j.trc.2025.105387
46. K. Li, H. Cui, Q. Meng, X. Zhang, Dynamic truck-drone collaborative transportation during hurricanes, *Transp. Res. Part C Emerg. Technol.*, **180** (2025) 105322. https://doi.org/10.1016/j.trc.2025.105322
47 Y. Shi, K. Zhang, J. Zhang, M. Chen, J. Yang, H. Guo, Dynamic truck-drone cooperative delivery of emergency supplies considering secondary disasters, *Transp. Res. Part E Logist. Transp. Rev.*, **206** (2026), 104543. https://doi.org/10.1016/j.tre.2025.104543
48. L. Costa, C. Contardo, G. Desaulniers, Exact Branch-Price-and-Cut Algorithms for Vehicle Routing, *Transp. Sci.*, **53** (2019), 946–985. https://doi.org/10.1287/trsc.2018.0878
49. S. A. Vásquez, G. Angulo, M. A. Klapp, An exact solution method for the TSP with Drone based on decomposition, *Comput. Oper. Res.*, **127** (2021), 105127. https://doi.org/10.1016/j.cor.2020.105127
50. M. F. Saeed Osman, Dynamic Computational Approach for Routing and Scheduling Last-Mile Distribution, *Int. J. Ind. Eng. Theory.*, **32** (2025). https://doi.org/10.23055/IJIETAP.2025.32.6.11095
51. A. Motaghedi-Larijani, Solving the number of cross-dock open doors optimization problem by combination of NSGA-II and multi-objective simulated annealing, *Appl. Soft Comput.*, **128** (2022), 109448. https://doi.org/10.1016/j.asoc.2022.109448
52. Y. Li, Z. Zhang, Q. Sun, Y. Huang, An improved ant colony algorithm for multiple unmanned aerial vehicles route planning, *J. Frankl. Inst.*, **361** (2024), 107060. https://doi.org/10.1016/j.jfranklin.2024.107060
53. K. Peng, J. Du, F. Lu, Q. Sun, Y. Dong, P. Zhou, et al., A Hybrid Genetic Algorithm on Routing and Scheduling for Vehicle-Assisted Multi-Drone Parcel Delivery, *IEEE Access*, **7** (2019), 49191–49200. https://doi.org/10.1109/ACCESS.2019.2910134
54. S. S. Özarık, V. Lurkin, L. P. Veelenturf, T. Van Woensel, G. Laporte, An Adaptive Large Neighborhood Search heuristic for last-mile deliveries under stochastic customer availability and multiple visits, *Transp. Res. Part B Methodol.*, **170** (2023), 194–220. https://doi.org/10.1016/j.trb.2023.02.016
55. S. Meng, Y. Chen, D. Li, The multi-visit drone-assisted pickup and delivery problem with time windows, *Eur. J. Oper. Res.*, **314** (2024), 685–702. https://doi.org/10.1016/j.ejor.2023.10.021
56. J. Cheng, Y. Zhou, C. Wu, Z. Li, Collaborative truck–drone–motorcycle delivery of emergency supplies for mountain wildfire suppression, *Comput. Ind. Eng.*, **196** (2024), 110468. https://doi.org/10.1016/j.cie.2024.110468

57. K. Li, T. Zhang, R. Wang, Deep Reinforcement Learning for Multiobjective Optimization, *IEEE Trans. Cybern*., **51** (2021), 3103–3114. https://doi.org/10.1109/TCYB.2020.2977661
58. Q. Guan, H. Cao, L. Jia, D. Yan, B. Chen, Synergetic attention-driven transformer: A deep reinforcement learning approach for vehicle routing problems, *Expert Syst. Appl*., **274** (2025), 126961. https://doi.org/10.1016/j.eswa.2025.126961
59. F. Akkerman, M. Mes, W. Van Jaarsveld, A comparison of reinforcement learning policies for dynamic vehicle routing problems with stochastic customer requests, *Comput. Ind. Eng*., **200** (2025), 110747. https://doi.org/10.1016/j.cie.2024.110747
60. Y. Zhou, W. Yang, Y. Gong, Reinforcement learning with priority decentralized PPO for multi-vessel cooperative rescue scheduling in flood disaster, *Alexandria Eng. J*., **138** (2026), 96–113. https://doi.org/10.1016/j.aej.2026.01.047
61. L. Du, X. K. Yang, Study on the impact of traffic and loss caused by different inundation depth, *Technological Innovation, Green Transportation: Proceedings of the 11th National Urban Road Transportation Academic Conference, Dalian: China Civil Engineering Society,* **5** (2011), 389–393.
62. A. Thibbotuwawa, G. Bocewicz, G. Radzki, P. Nielsen, Z. Banaszak, UAV Mission Planning Resistant to Weather Uncertainty, *Sensors*, **20** (2020), 515. https://doi.org/10.3390/s20020515
63. M. Tsang, D. M. Scott, An integrated approach to modeling the impact of floods on emergency services: A case study of Calgary, Alberta, *J. Transp. Geogr*., **86** (2020), 102774. https://doi.org/10.1016/j.jtrangeo.2020.102774
64. E. Al-Qadami, M. A. M. Razi, S. M. H. Shah, J. H. Pu, M. Amran, M. A. Díaz Huenchuan, et al., Floodwaters and vehicle hydrodynamics: A deep dive into risk mitigation unraveling vehicle stability in floods, *Results Eng*., **23** (2024), 102540. https://doi.org/10.1016/j.rineng.2024.102540
65. N. Agatz, P. Bouman, M. Schmidt, Optimization Approaches for the Traveling Salesman Problem with Drone, *Transp. Sci*., **52** (2018), 965–981. https://doi.org/10.1287/trsc.2017.0791
66. L. Zhen, J. Gao, Z. Tan, S. Wang, R. Baldacci, Branch-price-and-cut for trucks and drones cooperative delivery, *IISE Trans*., **55** (2023), 271–287. https://doi.org/10.1080/24725854.2022.2060535
67. J. Gao, L. Zhen, G. Laporte, X. He, Scheduling trucks and drones for cooperative deliveries, *Transp. Res. Part E Logist. Transp. Rev*., **178** (2023), 103267. https://doi.org/10.1016/j.tre.2023.103267
68. C. Liu, Z. Wu, Y. Zhou, L. Luo, X. Zhou, X. Xie, The optimization of dynamic and cooperative truck-drone distribution routes for emergency supplies delivery in pandemic areas, *J. Syst. Sci. Math. Sci*., **42** (2022), 3027–3043. https://doi.org/10.12341/jssms22266
69. X. Wang, L. Zhan, Y. Zhang, T. Fei, M. L. Tseng, Environmental cold chain distribution center location model in the semiconductor supply chain: A hybrid arithmetic whale optimization algorithm, *Comput. Ind. Eng*., **187** (2024), 109773. https://doi.org/10.1016/j.cie.2023.109773
70. M. Keskin, B. Çatay, Partial recharge strategies for the electric vehicle routing problem with time windows, *Transp. Res. Part C Emerg. Technol*., **65** (2016), 111–127. https://doi.org/10.1016/j.trc.2016.01.013
71. J. Lehmann, M. Winkenbach, A matheuristic for the Two-Echelon Multi-Trip Vehicle Routing Problem with mixed pickup and delivery demand and time windows, *Transp. Res. Part C Emerg. Technol*., **160** (2024), 104522. https://doi.org/10.1016/j.trc.2024.104522

72. M. Liu, Y. Li, X. Wang, Joint optimization of truck-drone routing for last-mile deliveries in urban areas, *Transportmetrica A: Transp. Sci*., (2024), 1–27. https://doi.org/10.1080/23249935.2024.2392611