Journal of Industrial and Management Optimization

JIMO, 22(5): 2347–2379.
DOI: 10.3934/jimo.2026086
Received: 27 January 2026
Revised: 13 March 2026
Accepted: 30 March 2026
Published: 14 April 2026

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

Research article

Dynamic optimization of multi-truck and multi-drone collaborative delivery for emergency material distribution

Yubo Sun[1], Weihua Liu[2,*], Jiaqin Hao[3] and Zhentao Shao[1]

[1] School of Management Engineering, Xuzhou University of Technology, Jiangsu, Xuzhou 221000, China
[2] College of Management and Economics, Tianjin University, Tianjin 300072, China
[3] School of Mathematics and Statistics, Suzhou University, Anhui, Suzhou 234000, China

* **Correspondence:** Email: lwhliu@tju.edu.cn; Tel: +8613512833463.

Abstract: Emergency material distribution, a crucial component of post-disaster emergency rescue, presents several challenges, including high distribution risk, complex road networks, halted traffic, and urgent demand. Even though vehicles have a large payload capacity, single-truck distribution models sometimes fall short of the strict timing requirements of emergency response. On the other hand, drone-based delivery systems show clear benefits in terms of accessibility, flexibility, and operational efficiency. The truck–drone synergistic distribution mode offers a unique means of significantly enhancing the efficiency of emergency supply delivery through resource integration and complementary advantages. To address this, a multi-vehicle, multi-drone, multi-circuit emergency material synergistic distribution path optimization model was created. The model aims to minimize the maximum delivery time among all trucks, while accounting for practical issues like the restriction of vehicle traffic following the disaster, the satisfaction of multiple demand points per flight, and the synergistic constraints of trucks and drones. As a solution to this model, the hybrid adaptive genetic algorithm (HAGA), incorporating dynamic programming, was proposed. The Solomon test case's R206, C206, and RC206 data sources were used to evaluate the algorithm's efficacy. Regardless of small, medium, or large-scale cases, the results demonstrate that the HAGA can greatly increase the efficiency of emergency material distribution following a disaster, with a maximum improvement of 15.41% in distribution efficiency and a significant improvement in computational efficiency, with a maximum reduction of 49.04% in computational time. According to the sensitivity analysis of the drone parameters, the optimal configuration of flight time and speed significantly improves the system performance. The proposed method provides insights for low-altitude economic applications and emergency logistics decision-making.

Keywords: emergency logistics; truck-drone collaborative system; dynamic collaboration model; path optimization strategy
Mathematics Subject Classification: 90B06, 90C59

1. Introduction

Natural disasters pose severe challenges to emergency logistics operations, including damaged road networks, transportation disruptions, and surging demand for relief supplies. When roads become impassable after disasters, truck-dependent transportation systems often fail to deliver essential supplies to affected communities in a timely manner, resulting in significant casualties and property losses. The 2008 Wenchuan earthquake, the 2021 Inner Mongolia floods, and the 2023 Jishishan earthquake in Gansu all underscore this predicament.

The rapid advancement of drone technology offers an innovative solution to this dilemma. Drones, with their exceptional accessibility and flexibility, can reach areas that are inaccessible to trucks [1,2]. However, their operational range remains relatively limited due to battery capacity constraints [3]. To overcome this limitation, a collaborative operation model between trucks and drones has emerged. This model integrates the complementary strengths of both transport modes. Trucks possess high payload capacity and long endurance, making them suitable for trunk-line transportation. Drones offer high flexibility and strong penetration capabilities, enabling last-mile delivery. Through resource integration and collaborative operations, overall transportation efficiency is significantly enhanced. The core logic of this collaborative model aligns with the principles of cooperative optimization, attracting increasing research attention [4]. With the rapid advancement of drone technology, their endurance and payload capacity continue to improve. A single drone flight can now complete multi-point deliveries, providing greater flexibility and speed for the truck–drone delivery collaboration model.

This study adheres to the fundamental principle of "time priority" in emergency management. Addressing the challenge of emergency supply distribution under restricted access conditions, it leverages the synergistic advantages of truck trunk-line transportation and drone last-mile delivery. A multi-vehicle, multi-drone, multi-loop collaborative distribution model is constructed, and the HAGA heuristic algorithm is designed for its solution. The objective is to meet the supply demands of all disaster-affected locations within the shortest possible time.

(1) Research on truck–drone coordinated delivery route planning

Since Wohlsen first proposed the concept of truck–drone collaborative delivery, this research field has attracted increasing attention from both academia and industry [5]. Companies such as Alibaba and JD.com have carried out extensive truck–drone collaborative delivery tests in urban and rural environments [6,7], while Meituan is currently exploring its application in food delivery services [8]. These practical explorations have stimulated significant academic research on truck–drone route optimization problems. Existing literature on truck–drone collaborative path planning can be categorized based on problem complexity, with its evolution progressing from single-truck–single-drone scenarios to more complex multi-truck–multi-drone configurations.

Early research focused on the single-truck–single-drone problem. Liu et al. pioneered mixed-integer programming model and a heuristic solution method aimed at minimizing the truck's return time to the depot [9]. Building upon this, Huang et al. extended the model by introducing two truck

behavior modes during drone flight: waiting or continuing along the predefined route [10]. Vásquez et al. further developed an exact Benders-type decomposition approach, combined with a two-stage mixed-integer programming paradigm, to achieve efficient coordination between the truck and drone routes [11]. Other researchers employed diverse solution techniques for single-vehicle–single-drone delivery path optimization, including mixed-integer linear programming based on the last-insertion algorithm [12], neighborhood search methods [13,14], and exact dynamic programming approaches suitable for large-scale instances [15,16]. While these studies established fundamental modeling frameworks for truck–drone coordination, they remain confined to single-truck scenarios and assume that drones serve only one customer per flight, failing to fully leverage the potential of modern drone technology.

With advances in drone technology, researchers have begun exploring configurations where a single truck carries and coordinates multiple drones, enabling simultaneous launch and oversight of multiple aircraft—a problem also known as the multi-drone traveling salesman problem [17,18]. For instance, Zeng et al. investigated path planning for fixed-position Multi-Unmanned Aerial Vehicle (UAV) deployment from a truck, requiring the truck to remain stationary until all deployed UAVs completed their missions and returned [19]. More recently, Peng et al. proposed a mixed-integer programming model allowing the simultaneous launch of multiple UAVs from a truck, supporting multiple round trips to customer sites per UAV. This model employs a branch-and-bound algorithm based on finite neighborhood search [20]. Building upon this foundation, Luo et al. further extended the research by permitting drones to take off and land at any point along the truck's route and designed an exact branch-and-bound algorithm based on column generation [21]. While these studies significantly enhance the operational flexibility of truck–drone systems, they remain confined to single-truck scenarios and often assume idealized operating conditions, neglecting real-world constraints such as road disruptions.

(2) Studies on cooperative emergency logistics distribution using trucks and drones

In commercial logistics scenarios, truck–drone collaboration systems demonstrate significantly superior performance compared to standalone systems. Commercial logistics focuses more on balancing cost and efficiency. In contrast, disaster relief operations prioritize timeliness and require the rapid delivery of critical supplies such as food, drinking water, and medicine to victims in life-threatening situations [22]. Faced with the dual pressures of a drastically reduced road transport capacity and surging rescue demands, this truck–drone collaboration has become the preferred strategy for rapid emergency logistics distribution.

To explore how to fully leverage the advantages of trucks' high payload capacity and drones' flexibility, Amirsahami et al. established a contactless medical supply delivery system based on truck-drone collaboration, which significantly enhanced delivery efficiency and coverage [23]. Duan et al. proposed using trucks as mobile warehouses to support drones in rapid last-mile delivery through continuous task allocation and dynamic resource reallocation [24]. These studies represent preliminary explorations into the feasibility and fundamental operational models of truck–drone collaborative emergency logistics. To address the timeliness of material delivery, Rahimi et al. developed a hybrid collection-dispatch optimization model based on truck-drone collaboration to reduce post-earthquake rescue costs and solved it with an improved adaptive large neighborhood search algorithm [25]. For the real-world scenario of the Sichuan earthquake, Shi et al. applied a mixed-integer linear programming model to solve the emergency material dispatch coordination scheme for electric vehicles and drones [26].

Timeliness remains the primary objective in emergency supply delivery. Zhang et al. constructed an integer programming model that integrates dynamic programming techniques to meet the demands of disaster-stricken areas [27], while Mulumba et al. employed a segmented delivery strategy to build a multi-vehicle, multi-drone, multi-trip mixed-integer programming model [28]. Both effectively addressed the efficiency challenges in truck–drone collaborative emergency supply delivery. Sun et al. enhanced rescue efficiency by proposing a priority-weighted delayed-cost reduction framework within truck-drone coordination systems [29]. Although these models increase the technical complexity of emergency logistics optimization, they usually assume idealized operating conditions and overlook the unique challenges presented by post-disaster scenarios with restricted road access, which is a critical requirement for emergency supply distribution.

Consequently, recent literature has begun to explore how to optimize truck–drone delivery routes in complex road network environments characterized by damaged roads and restricted access. Yan et al. divided disaster areas into restricted zones, no-fly zones, and normal zones, employing shortest path methods and tabu search algorithms to solve the delivery route optimization problem for a single truck carrying one drone serving multiple demand points [30]. Zhang et al. addressed emergency supply scheduling with road damage and congestion, where each truck carries only one drone [31]. Lu et al. proposed optimizing drone delivery routes for demand points within waterlogged obstacle zones using a single truck carrying multiple drones [32]. Hu et al. introduced the initial emergency response multi-truck and multi-drone delivery path problem, where a single drone launch can serve multiple nodes [33]. Liu et al. addressed the optimization problem of coordinating emergency supply routes under constrained access conditions, where demand is decomposable, and a single drone deployment can serve multiple nodes [34]. Circular regions are employed to represent areas rendered inaccessible by disasters such as earthquakes or floods, with each circle's radius corresponding to the severity of infrastructure damage. However, the model assumes drones can be launched and received at any point along the truck's route, which may pose practical challenges in real-world operations.

The above studies indicate that to address emergency supply distribution in specific scenarios, the constructed optimization models have become increasingly complex, making their solution more challenging. To overcome computational bottlenecks, this paper employs the variable-dimensional particle swarm optimization heuristic algorithm proposed in [35], successfully resolving the two-layer traveling salesman problem (TSP) for humanitarian logistics in truck–drone collaborative environments. This approach comprehensively considers distribution center capacity constraints, time window limitations, and random demand characteristics. The dynamic truck–drone coordination framework proposed in [36] deepens the theoretical understanding of multi-drone and multi-vehicle scheduling. However, this complex model faces computational bottlenecks when dealing with large-scale scenarios. Amirsahami et al. integrated fuzzy programming with the non-dominated sorting genetic algorithm II (NSGAII) to solve the positioning problem of drones and dispersed facilities in urban rescue chains, achieving a 75% cost reduction and shorter waiting times [23]. For obstacle avoidance scenarios, Lu et al. proposed a multi-drone–truck coordination scheme based on a simulated annealing-whale hybrid algorithm, which improved obstacle coverage by 35% [32]. However, it is still limited to single-vehicle route planning. Liu et al. designed a variable neighborhood search (VNS) algorithm to solve a multi-vehicle, multi-drone, multi-loop emergency supply coordination path optimization model for demand-decomposable scenarios with restricted access, where a single drone launch serves multiple nodes [34]. Hu et al. addressed the multi-truck–multi-drone delivery path problem using a VNS algorithm [33]. However, this algorithm shows exponential growth of

computational time with the problem dimension. That is, it requires a large number of iterations for large-scale problems, which may result in extended computation time. These studies indicate that while truck–drone collaborative delivery models theoretically offer "minute-level" response capabilities in damaged networks, existing algorithms still face significant challenges in simultaneously meeting the robustness, computational efficiency, and solution quality required for real-time emergency logistics.

In conclusion, existing research has laid a solid foundation for in-depth studies on post-disaster emergency supply delivery routes. However, further exploration is needed in the following areas: (1) All the aforementioned literature focuses on single-point delivery during a single drone flight. However, with the advancements in drone technology, their payload capacity and endurance have significantly improved, enabling multi-point delivery in a single flight, which is particularly suitable for small, lightweight items such as emergency medicines and blood supplies. (2) At the algorithmic level, most studies focus on single-vehicle–single-drone or single-vehicle–multiple-drone delivery scenarios, while research on multi-vehicle–multi-drone–multi-loop delivery path algorithms remains limited. (3) Existing literature predominantly examines truck–drone collaborative delivery under ideal road network conditions. However, the most typical application scenario for truck–drone collaboration is emergency supply delivery during natural disasters when vehicle access is obstructed. Only a few studies have explored multi-vehicle, multi-drone collaborative delivery under traffic-restricted conditions, but their model assumptions and algorithms require further optimization. This research explores the multi-truck, multi-drone collaborative path problem under constrained conditions. It allows drones to serve multiple customers per flight and focuses on the unique demands of post-disaster road access limitations.

Based on the above analysis, this paper investigates the path optimization problem for multi-truck–multi-drone collaborative emergency supply delivery under restricted access conditions. In this scenario, demand points in unrestricted areas are served by trucks traveling along circular routes that cover multiple demand points. Meanwhile, demand points in restricted areas are delivered to by vehicle-mounted drones, and each drone is capable of consecutively servicing multiple restricted demand points per flight. This study constructs a multi-truck–multi-drone collaborative delivery path planning model aimed at minimizing the total emergency supply delivery time. Based on the model's characteristics, HAGA, incorporating multiple improvement strategies, is designed as the solution approach. Numerical experiments demonstrate that this algorithm rapidly obtains high-quality solutions, significantly enhancing emergency logistics delivery efficiency. The main contributions of this paper are as follows:

(1) This study proposes a more universally applicable collaborative model of "multiple trucks, multiple drones, and multiple loops delivery". This model enables drones to consecutively serve multiple demand points in restricted-access zones during a single launch while accommodating multi-loop truck deliveries in unrestricted areas. This configuration better aligns with the practical requirements for the bulk, multi-point delivery of lightweight goods such as emergency medical supplies, theoretically enriching the operational scope of truck–drone collaborative delivery.

(2) This study addresses the combined optimization characteristics of multiple trucks, multiple drones, and multiple loops by designing HAGA with a self-calibration operator. This algorithm not only enhances search efficiency and solution quality within complex solution spaces but also provides a new methodological reference for solving such high-dimensional vehicle–drone path planning problems.

(3) This study uses "access restrictions" as the core constraint, strictly classifying demand points into unrestricted zones and restricted zones. Then, it assigns tasks to trucks and drones accordingly, ensuring that the model accurately captures the transportation characteristics during the initial stages of disasters such as earthquakes and floods. The findings provide scientific decision-making support for emergency management departments in formulating coordinated truck–drone delivery plans, offering direct practical guidance for enhancing the timeliness and reliability of post-disaster emergency supply responses.

The remainder of this paper is organized as follows: Section 1 presents the research background, highlighting the challenges of post-disaster emergency logistics and the potential of truck–drone collaborative systems. It establishes the theoretical foundations of this study and articulates the research objectives and contributions. Section 2 provides a comprehensive review of the relevant literature on truck–drone collaborative routing, with an emphasis on studies in both commercial logistics and emergency logistics contexts. It also identifies the research gaps that motivate this study. Section 3 describes the research problem in detail, including the problem description, model assumptions, and the mathematical formulation of the multi-truck–multi-drone collaborative routing model under access constraints. Sets, parameters, and decision variables are defined separately. Section 4 presents HAGA, which is designed to solve the proposed model. This includes the adaptive mechanisms for crossover and mutation probabilities, the integration of dynamic programming for initial solution generation, and detailed explanations of all algorithmic components. Section 5 reports numerical experiments based on Solomon benchmark instances. It compares the performance of HAGA with genetic algorithm (GA) and VNS algorithms across small, medium, and large-scale scenarios, followed by a sensitivity analysis of key drone parameters. Section 6 concludes the paper with a summary of findings, theoretical and practical contributions, management insights, and directions for future research.

2. Problem description

2.1. Problem statement

Emergency supplies, including medications and everyday essentials, must be distributed as quickly as possible to all impacted demand locations during significant pandemics or natural catastrophes. Certain impacted locations are unreachable to trucks due to restricted road accessibility in disaster-affected areas, requiring delivery by truck-mounted drones. Delivering supplies to various impacted locations is the responsibility of the emergency supplies distribution center, which is outfitted with numerous trucks and unmanned aerial vehicles. Demand points are divided into a set of truck-accessible locations and a set of truck-inaccessible points (i.e., drone-exclusive points) based on information about road conditions and crisis situations. To improve delivery efficiency, vehicles service truck-accessible locations, while drones fulfill demand at truck-inaccessible locations. Trucks leave the distribution center carrying goods while carrying multiple drones, with the demand at each affected point not exceeding the drone payload capacity limit, based on information on road distances and damage conditions in each disaster zone, locations of emergency supply demand points, demand quantities at each point, service times, truck speed and payload capacity, and drone flight speed and payload capacity. To cut down on overall truck delivery time, trucks at drone launch locations first launch drones before completing their own maintenance tasks. They then proceed to the next affected

area. Within the limitations of their payload and range, drones transport cargo to several areas close to the launch point that are inaccessible by trucks. The vehicle completes its servicing task at the receiving area prior to retrieving the drones. When the commencement time is set to zero, every car and drone leaves the distribution center simultaneously. As shown in Figure 1, both trucks and drones must return to the distribution center after finishing delivery missions at all impacted locations.

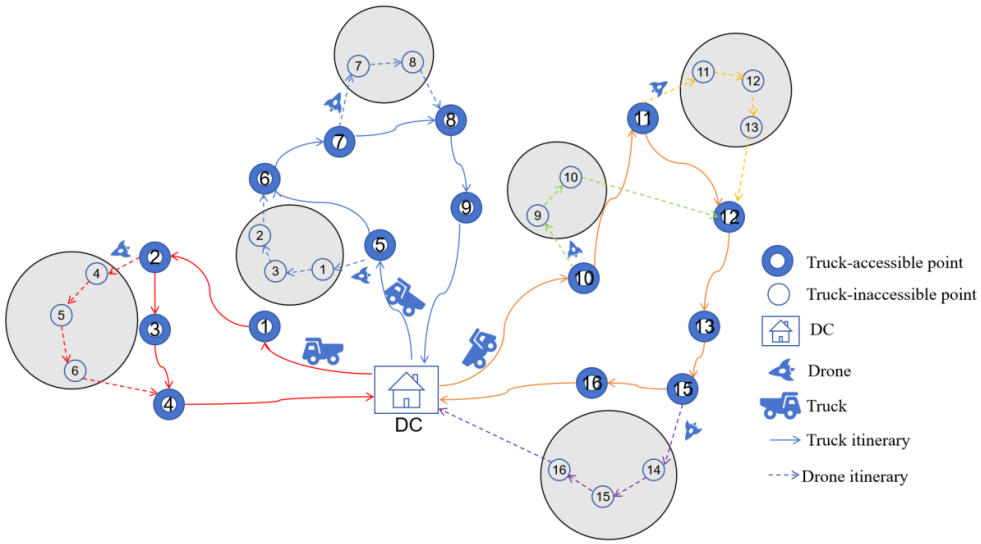

Figure 1. Dynamic coordinated emergency supply distribution via multi-truck–multi-drone. Note: Solid lines indicate truck routes, while dashed lines indicate drone flight paths. The truck departs from the distribution center and launches the drone at demand point ② in the general area. After the drone consecutively serves demand points ④⑤⑥ in multiple restricted-access zones, it returns to rendezvous with the truck at the subsequent service point ④. Following drone launch, the truck continues forward to serve other demand points in the general area, enabling simultaneous vehicle–drone operations. Multiple trucks and drones operate concurrently to collectively cover all demand points, demonstrating dynamic coordination across three levels: task allocation, launch/recovery, and parallel operations.

2.2. Model assumptions

The following assumptions are made for the model in order to define the extent of the research problem:

(1) A truck or a drone must visit each disaster-affected area precisely once. Truck-accessible points and truck-inaccessible points are only supplied by trucks and drones, respectively, to reduce the operational dangers and scheduling complexity of drones.

(2) The demand at each disaster-affected point is less than the load capacity of both trucks and drones. Multiple demand points can be served by a single drone deployment; trucks can launch and recover multiple drones at a single service station; and drones are not allowed to land on trucks other than their launch platform.

(3) Each truck can carry multiple drones. Drones can only be launched and retrieved at affected points or the distribution center and cannot be launched or retrieved while the truck is in motion.

(4) Drones are launched, and a truck moves on to the next available disaster-affected area. The drone returns to meet the truck at its current or next service stop after finishing its delivery mission from a single launch, when its battery is changed for the subsequent mission; the battery swap time is regarded as insignificant.

(5) Drones and vehicles both move at steady speeds. Every truck has the same maximum load capacity and is not limited by trip distance. Each drone is of the same model, can be launched more than once, and has a maximum range determined by battery capacity.

To ensure that all disaster-affected areas receive the necessary supplies while achieving the fastest emergency response time and the maximum distribution resource utilization rate, the following elements need to be taken into account while designing truck and drone routes: 1) all disaster-affected locations must be serviced, and 2) the overall time needed to finish delivery to all impacted locations must be reduced. As such, how can we dynamically plan coordinated distribution routes for trucks and drones to reduce the overall emergency material distribution time, given that all trucks and drones leave the distribution center and have to return to their original depot after servicing all disaster-affected locations?

3. Model construction

3.1. Notations and variable description

The sets, parameters, and decision variables in the model are listed in Tables 1, 2, and 3, respectively.

Table 1. Definition of sets.

Symbol	Definition
$C_k = \{1,2,\cdots, k\}$	The set of disaster-affected points that can be reached by trucks
$C_f = \{1,2,\cdots, f\}$	The set of disaster-affected points that cannot be reached by trucks
$C = C_k \cup C_f \cup \{0, k+f+1\}$	The complete set of nodes, where 0 and c+1 denote the distribution center
G	The set of arcs, which are pairs of nodes (i, j)
K	The set of trucks available for the operation
F	The set of drones deployed in the mission
Ω	The set of launch points
$\mathfrak{R}$	The set of receiving points

Table 2. Definition of parameters.

Symbol	Definition
G_f	The maximum payload capacity of the drone
g_f	The curb weight of a single drone
v_k	The traveling speed of the truck
v_f	The cruising speed of the drone
L_f	The maximum flight endurance range of the drone

Table 3. Definition of variables.

Symbol	Definition
k	The number of trucks
f	The number of drones
f_k	The number of drones carried by truck k, $i \in C_k, k \in K$
q_i	The relief demand at disaster-affected node i, $i \in C_k \cup C_f$
s_i	The service time required at disaster-affected node i, $i \in C_k \cup C_f$
R_i	The congestion radius of blocked region i, $i \in C_f$
s_i^k	The service time of truck k at disaster-affected node i, $i \in C_k, k \in K$
s_i^f	The service time of drone f at disaster-affected node i, $i \in C_f, f \in F$
d_{ij}	The Euclidean distance between node i and node j, $i, j \in C, i \neq j$
t_{ij}^k	The time required for truck k to traverse the arc (i, j), $i, j \in C_k, k \in K$
t_{ij}^f	The flight duration of drone f along the arc (i, j), $i, j \in C_f, f \in F$
η_i^k	The specific moment when truck k sets off from node i, $i \in C_k, k \in K$
η_i^f	The specific moment when drone f sets off from node i, $i \in C_f, f \in F$
t_i^k	The elapsed time starting from the moment when truck k departs from the distribution center and ending at the time when it finishes its service at node i, $i \in C_k, k \in K$
t_i^f	The time elapsed from the moment when drone f departs from the distribution center until it finishes its service at node i, $i \in C_f, f \in F$
ψ_i^f	The specific moment when drone f takes off from node i, $i \in C_f, f \in F$
π_i^f	The recovery duration of drone f at node i, $i \in C_f, f \in F$
θ_i^k	The duration for which the truck waits at node i, $i \in C_k, k \in K$
θ_i^f	The duration for which the drone waits at node i, $i \in C_f, f \in F$
x_{ij}^k	0-1 variable equal to 1 if truck k traverses arc (i, j) (i.e., departs from node i and arrives at node j), and 0 otherwise, where $i, j \in C_k, k \in K, i \neq j$
x_{ij}^f	0-1 variable equal to 1 if drone f traverses arc (i, j) (i.e., launches from node i and flies to node j), and 0 otherwise, where $i \in C_k, j \in C_f, f \in F, i \neq j$
x_i^k	0-1 variable equal to 1 if truck k serves disaster-affected node i, and 0 otherwise, where $i \in C_k, k \in K$
x_i^{kf}	0-1 variable: 1 if drone f on truck k services disaster point i, 0 otherwise, where $i \in C_f, f \in F$
x_{ij}^{kf}	Variable 0-1: If drone f on truck k flies along arc (i,j), i.e., the drone launches from point i and flies to point j, it is 1; otherwise, it is 0; where $(i, j) \in G, k \in K, f \in F, i \neq j$
t_{ij}^{kf}	The flight time of drone F along arc (i, j) on truck k, where $(i, j) \in G, k \in K, f \in F, i \neq j$
y_{ij}^{kf}	0-1 variable: set to 1 if drone f on truck k is launched at point i and retrieved by the truck at point j; otherwise, set to 0; where $(i, j) \in G, k \in K, f \in F, i \neq j$
L_i^f	The remaining flight range of drone f upon departing node i, $i \in C_f, f \in F$
y_i^{kf}	0–1 variable equal to 1 if truck k launches drone f at disaster-affected node i, and 0 otherwise, where $i \in C_k, k \in K$, $f \in F$.
z_i^{kf}	0–1 variable equal to 1 if truck k recovers drone f at disaster-affected node i, and 0 otherwise, where $i \in C_k, k \in K$, $f \in F$.

3.2. Model formulation

3.2.1. Objective function

During catastrophes, emergency logistics distribution activities must meet all disaster-affected points' needs as soon as possible. Consequently, the objective function of the model is set to minimize the time taken by the truck that takes the longest time to complete the distribution task of all affected points, which is [31]

$$\min Z = \max\{t^k_{k+f+1}\}_{i\in K} \tag{1}$$

3.2.2. Constraints

(1) Vehicle limitations

In this section, we delve into the specific limitations of the vehicle. These limitations significantly impact the performance, functionality, and overall operation of the vehicle.

$$\sum_{k\in K} x^k_j = 1 \qquad (\forall j \in C_k) \tag{2}$$

$$\sum_{i\in C_k} x^k_{0i} = \sum_{j\in C_k} x^k_{j0} \qquad (\forall k \in K) \tag{3}$$

$$\sum_{i\in C_k} x^k_j \cdot q_i + \sum_{f\in F}(\sum_{i\in C_f} x^f_i \cdot q_i + f_k \cdot g_f) \le G_k \qquad \forall k \in K \tag{4}$$

$$\sum_{i\in C_k\cup 0} x^k_{im} = \sum_{j\in C_k\cup 0} x^k_{mj} \qquad \forall m\in C_k, \forall k\in K \tag{5}$$

$$t^k_j = (\eta^k_i + \frac{d_{ij}}{v_k})x^k_{ij} \qquad \forall i,j \in C_k \cup 0, i\neq j, \forall k\in K \tag{6}$$

$$\eta^k_i = t^k_i + s^k_i + \theta^k_i \qquad \forall i\in C_k\cup 0, \forall k\in K, \forall \theta_k \ge 0 \tag{7}$$

Equation (2) represents that each truck-accessible point must be served exactly once by a truck. Equation (3) represents that the delivery route of each truck must start and terminate at the distribution center. Equation (4) ensures that the total weight of the truck along with all loaded equipment, including drones and materials, does not exceed its maximum payload capacity. Equation (5) represents the flow-conservation constraint, requiring that a truck entering a node must subsequently depart from it. Equation (6) defines the arrival time of the truck at a given node. Equation (7) defines the departure time of the truck from a given node.

(2) Drone constraints

$$x^{kf}_j(\sum_{i\in C_k} x^{kf}_{ij} - 1) = 0 \qquad \forall j \in C_f, k \in K, f \in F \tag{8}$$

$$x_j^{kf}\left(\sum_{i\in C_k} x_{ji}^{kf} - 1\right) = 0 \qquad \forall j \in C_f, k \in K, f \in F \tag{9}$$

$$\sum_{j\in C_f} x_{ij}^{f} \cdot q_j \le G_f \qquad \forall f \in F, \forall i \in C_k \tag{10}$$

$$\sum_{i\in C_k \cup C_f \cup 0} x_{im}^{f} = \sum_{j\in C_k \cup C_f \cup 0} x_{mj}^{f} \qquad \forall m \in C_f, \forall f \in F \tag{11}$$

$$L_i^f = L_f \cdot y_i^{kf} \qquad \forall k \in K, \forall f \in F, \forall i \in C_k \cup 0 \tag{12}$$

$$L_j^f = L_i^f - d_{ij} \cdot x_{ij}^f \qquad \forall f \in F, \forall i, j \in C_k \cup C_f \cup 0 \tag{13}$$

$$t_j^f = (\eta_i^f + \frac{d_{ij}}{v_f}) x_{ij}^f \qquad \forall i, j \in C_k \cup C_f \cup 0, i \ne j, \forall f \in F \tag{14}$$

$$\eta_i^f = t_i^f + s_i^f \cdot x_i^f \qquad \forall i \in C_f, \forall f \in F \tag{15}$$

$$(L_i^f - d_{ij} \cdot x_{ij}^f) \cdot z_j^{kf} \ge 0 \qquad \forall k \in K, \forall f \in F, \forall i \in C_f, j \in C_k \cup 0 \tag{16}$$

Equations (8) and (9) represent the drone traffic conservation constraint, meaning that after delivering emergency supplies to disaster-stricken point j, the drone must depart from point j. Inequality (10) indicates that the total demand for all disaster-stricken points served by drone f launched from node i must not exceed the drone's maximum payload capacity. Equation (11) imposes the flow-conservation constraint, requiring that a drone departing from any affected area must be the same one that entered it. Equation (12) specifies that the drone's battery must be fully charged at the moment of each launch. Equation (13) defines the remaining flying range of the drone after it travels from node i to node j. Equation (14) defines the drone's arrival time at a node. Equation (15) defines the drone's departure time from distribution node i. According to inequality (16), the remaining flight range of drone f after returning from node i to the truck service station j must be non-negative.

(3) Truck–drone coordination constraints

$$\sum_{i\in C_f} x_{mi}^f = \sum_{j\in C_f} x_{jl}^f \qquad \forall m, l \in \{0\} \cup C_k, l \ge m, \forall f \in F \tag{17}$$

$$\psi_j^f = t_j^k \cdot y_j^{kf} \qquad \forall i \in C_k \cup \{0\}, \forall k \in K, \forall f \in F \tag{18}$$

$$\pi_j^f = (t_j^k + s_j^k) \cdot z_j^{kf} \qquad \forall i \in C_k \cup \{0\}, \forall k \in K, \forall f \in F \tag{19}$$

$$\theta_i^k = \max(((t_j^k + s_j^k) - (\eta_i^f + {}^{d_{ij}}\!/_{v_f})) \cdot z_j^{kf}, 0) \qquad \forall j \in C_k, \forall i \in C_f, \forall k \in K, \forall f \in F \tag{20}$$

$$\theta_i^f = \max(((\eta_i^f + {}^{d_{ij}}\!/_{v_f}) - (t_j^k + s_j^k)) \cdot z_j^{kf}, 0) \qquad \forall j \in C_k, \forall i \in C_f, \forall k \in K, \forall f \in F \tag{21}$$

$$\theta = \sum_{i,j \in C_k} \sum_{k \in K} \sum_{f \in F} (\theta_i^k + \theta_j^f) \tag{22}$$

Equation (7) dictates that a drone launched from the distribution center or a truck must return to the distribution center or truck after completing its delivery to drone-exclusive nodes. Equation (18) ensures that the launch time of the drone from node j is synchronized with the truck's arrival time at that node. The drone recovery time at node j is defined by equation (19) as the total of the truck's service time and arrival time at node j. Equation (20) defines the waiting time of trucks at node j for drone landing. The drone's waiting period for truck availability at node j is defined by equation (21). The entire waiting time resulting from truck–drone synchronization is measured by equation (22).

(4) Other variable constraints

$$x_{ij}^k \in \{0,1\},\ x_{ij}^f \in \{0,1\},\ x_i^k \in \{0,1\},\ x_i^f \in \{0,1\},\ y_i^{kf} \in \{0,1\},\ z_i^{kf} \in \{0,1\} \tag{23}$$

The decision variables are subject to domain limits by equation (23).

4. Algorithm design

In this scenario, the distribution center can send out several vehicles at once, each carrying several drones, to carry out delivery duties along various routes. As a result, a vehicle routing problem (VRP) can be used to simulate the truck routing problem. A drone operating from a truck stop, repairing impacted locations, and then returning to a truck or the distribution center is similar to a TSP. Due to its complexity, the problem as a whole is categorized as NP-hard, since it is a very complicated combinatorial optimization problem. This paper suggests HAGA, combining dynamic programming and GA concepts based on the features of the problem and model. Below is a presentation of the comprehensive algorithmic framework:

(1) Drone launch site selection and task distribution: Select drone launch sites from truck-accessible places and assign individual drones delivery missions for non-truck-accessible points. (2) Chromosome encoding: To sequentially encode the order of truck and drone node visits, use a two-dimensional integer encoding approach. (3) Initial population construction: Randomly generate initial truck delivery routes using a greedy strategy, then solve drone delivery routes via integer programming. (4) Fitness function design: Based on the node sequence along the truck delivery route, use the nearest neighbor node algorithm to determine the initial drone delivery path. (5) Selection operation: To preserve population quality and speed up the optimization process, choose individuals with greater fitness to join the next generation. (6) Crossover: This technique effectively explores potential optimal solutions in multi-truck, multi-drone emergency supply route optimization situations by generating offspring by mixing information from two or more current people. (7) Mutation operation: Randomly alters portions of genes within an individual's chromosome to introduce novel solutions. The population size increases when created individuals are subjected to mutation. This method improves overall optimization performance and speed by striking a balance between local and global search capabilities. Figure 2 shows the specific algorithm.

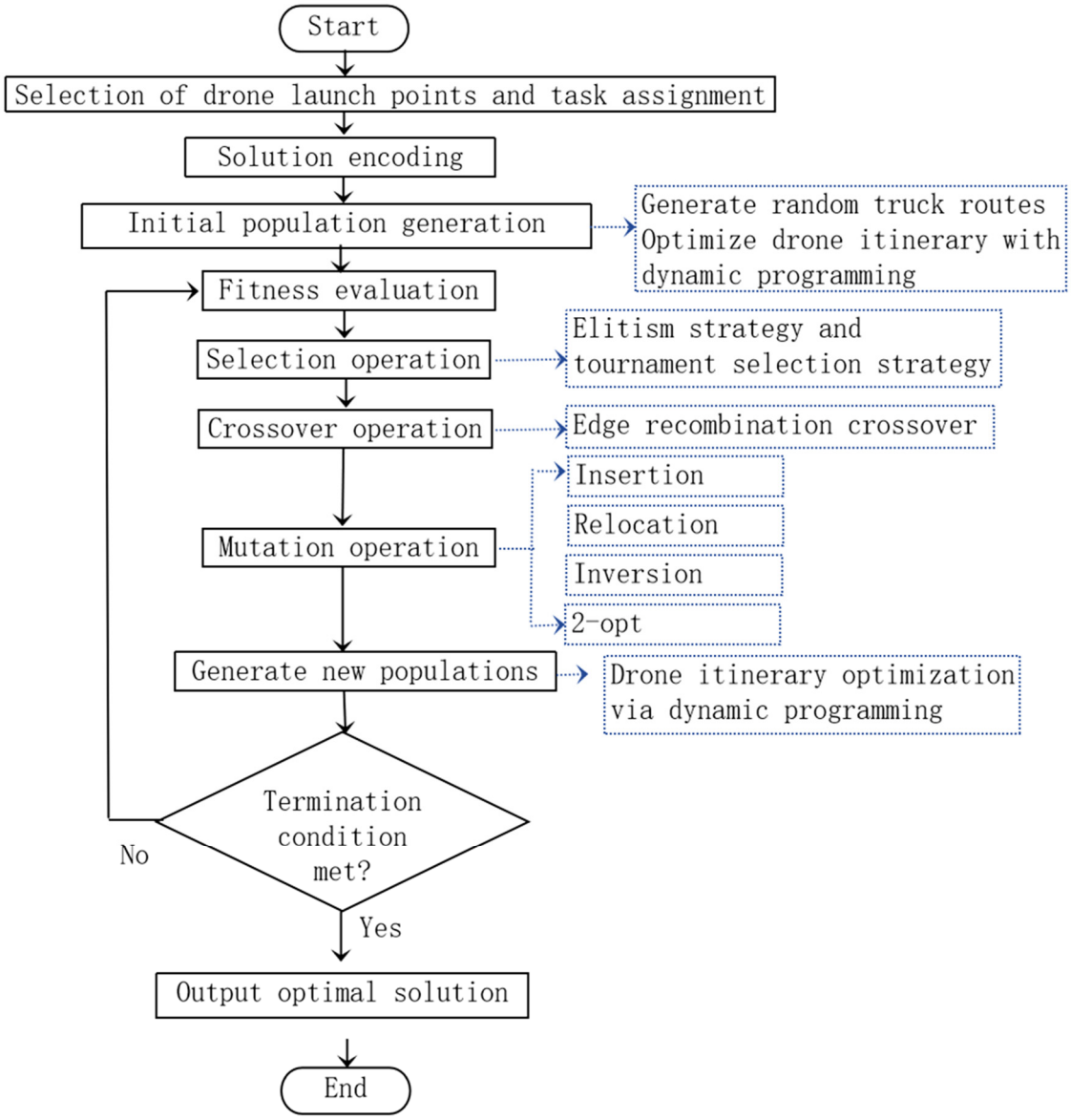

Figure 2. Flowchart of the HAGA for the multi-truck and multi-drone system.

4.1. Selection of drone launch points and task allocation

Determine the separation between each drone's designated point and every truck-accessible point. Choose the closest location that can be reached by the vehicle to launch the drone. Depending on how many drone task points each launch point is in charge of, task allocation can be split into two scenarios:

(1) Each task point is allocated to one drone, meaning that one drone serves just one task point, if the number of drone task points given to a launch site does not exceed the number of drones transported by the truck.

(2) If the number of task points assigned to a launch point exceeds the number of drones carried by the truck, some drones must serve multiple task points. In this case, generate a set of drone delivery task points while satisfying the constraints of drone payload capacity and endurance range.

4.2. Chromosome encoding

The following structure is used in a two-dimensional integer encoding scheme: The truck's delivery route is shown in Part 1, which lists the node indices it visited in order. In this case, the distribution center is indicated by 0. The choice of drone launch locations is shown in Part 2, which is made up of 0s and 1s, where 1 denotes a drone launch site, and 0 denotes one that is not. Figure 3 illustrates the chromosomal encoding structure using the route taken by three vehicles visiting 16 catastrophe areas as an example.

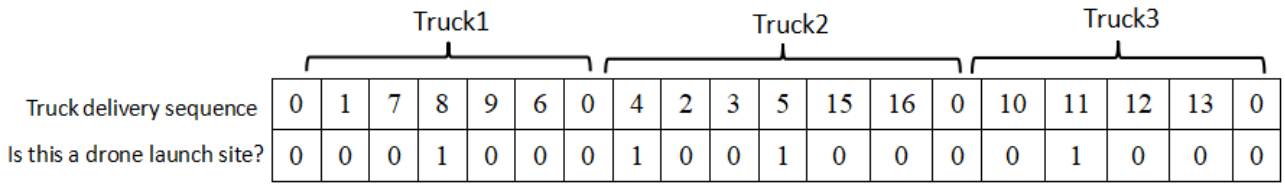

Figure 3. Schematic diagram of solution encoding.

4.3. Construction of the initial population

Program runtime can be shortened, and optimal or adequate answers can be obtained quickly with high-quality initial solutions. Since the delivery locations of both trucks and drones are known, the key to solving the multi-truck–multi-drone dynamic collaborative delivery problem is figuring out the launch and reception locations of the drones as well as scheduling their wait times. This study suggests a two-stage initial solution generation mechanism to handle this: (1) Use heuristic algorithms to quickly create the initial truck delivery routes; (2) use dynamic programming to create accurate drone delivery routes to create a high-quality initial population.

(1) Initial truck routes

The initial truck route constitutes a classic VRP problem. To prevent premature convergence of the algorithm, the initial population construction strategy employs greater diversity: 40% of solutions utilize a universal heuristic algorithm designed with a greedy approach to determine the initial delivery routes for vehicles, and 60% of solutions are randomly generated, ensuring diversity within the initial population. The steps for designing truck routes using the greedy approach are as follows:

a. Initialization preparation: Randomly rearrange all truck-accessible points to form a set of accessible points C_k, add all these points to the unassigned set $U_k = C_k$, and initialize an empty truck path set $T_T = \{\}$.

b. Initialize truck route: Starting from distribution center 0, initialize a current route $r = \{0\}$ and set the truck's initial remaining capacity.

c. Select the next truck-accessible point: For the final node of the current path, compute its distance to all unassigned truck-accessible points $i \in U_k$. Under the constraint $q_i \le q_k$ (truck capacity), select the nearest truck-accessible point i^*. If no qualifying truck-accessible point exists, terminate the current truck path and return to distribution center 0.

d. Update path: Add the selected truck-accessible point i^* to the current path r, where $r = \{0, i^*\}$. Update the truck's remaining capacity $q_{k+1} = q_k - q_i^*$. Remove the affected point i^* from the unassigned set U_k.

e. Completion path: If all reachable points have been assigned or the current vehicle capacity is insufficient, return the current path to distribution center 0, i.e., $r = \{r, 0\}$, add this path r to the path set R_T and then initialize a new path and return to step ②.

f. Termination: When the unassigned set $U_k = \phi$, the algorithm terminates and returns the path set R_T, thereby generating an initial solution for the truck delivery routes.

(2) Initial drone route generation

Since drones only serve a limited number of delivery points within the truck-restricted zone, this paper employs an exact algorithm to generate the initial drone delivery routes.

Considering the distribution of delivery points, drone route planning is divided into the following two scenarios:

a. When the number of demand points does not exceed the number of drones, the corresponding number of drones can be dispatched directly from the launch point to perform delivery tasks, with each drone serving only one demand point.

b. When the number of delivery points exceeds the number of drones, a single truck may need to carry multiple drones to complete the delivery task. Since the time drones spend waiting for trucks does not affect the objective function, this paper disregards it and only accounts for the time cost incurred by trucks waiting for drones. In this scenario, dynamic programming is employed to plan the drone delivery routes. Truck waiting time $\theta_i^k = \max\left\{0, t_i^k - \pi_i^f\right\}$.

Steps for dynamic path planning of drones:

Step 1: Evaluate feasible paths and calculate their costs. For any task subset S and receiving point i, compute the total flight distance $D(S,i)$ required for the drone to complete that task set. The collaborative time $w(S,i)$ generated by the drone waiting at receiving point r is calculated as follows: If the path satisfies constraints such as drone endurance and payload capacity, (S,i) is considered a feasible combination, and its corresponding waiting time θ_i^k is computed.

Step 2: Establish a dynamic programming state transition model. Use a bitmask to represent the set of completed tasks; $f(mask)$ denotes the minimum maximum waiting time when the task set mask is completed, and $A(mask)$ denotes the task reception point allocation scheme corresponding to the mask. For any state mask, enumerate all feasible task subsets S and reception points i. If combination (S,i) is a feasible scheme, transition from state mask\S to state mask and update the optimal value according to the following formula:

$$f(mask) = \min_{S \subseteq mask, i \in R, (S,i) \in \Omega} \max\left\{f(mask \setminus S), w(S,i)\right\} \tag{24}$$

where R denotes the set of receiving points, Ω represents all feasible combinations satisfying the constraints, and the initial state must meet the corresponding conditions $f(0) = 0$.

4.4. Fitness function design

In the HAGA for truck route planning, a fitness function is designed based on the optimization objective to evaluate each individual's performance within the problem domain. The objective is to minimize the total time spent by the truck with the longest delivery time to all disaster-affected points.

The fitness function can be designed as a negative objective value: $Fitness\quad Function = -\max\left\{t_i^k\right\}_{i\in C_k}$. In order to deal with constraint violations and make sure that those who do so are excluded during selection, a very large penalty factor M is added. For instance, if a delivery plan violates constraints such as drone flight distance limits or truck load capacity limits, M is added to its fitness value: $Fitness\quad Function = -\max\left\{t_i^k\right\}_{i\in C_k} - M\cdot\sum_i violations_i$, where $violations_i$ denotes the number of violations of constraint i, and M is a sufficiently large integer.

4.5. Adaptive crossover and mutation probabilities

In genetic operations, the crossover probability P_c and mutation probability P_m are critical to algorithm performance. Traditional genetic algorithms employ fixed probabilities, which can easily destroy high-quality solutions in later generations [37,38]. In light of this, the proposed HAGA algorithm dynamically adjusts crossover and mutation probabilities based on changes in individual fitness values across different evolutionary stages.

(1) Evolution stage division

Let g denote the current evolution iteration and $G_{\max}$ the maximum iteration count. The evolution stage is defined as:

$$P_{\mathrm{g}} = \frac{g}{G_{\max}} \tag{25}$$

Stage-dividing function:

$$S(P_{\mathrm{g}}) = \begin{cases} \exp loration & , p_g < 0.3 \\ transition & ,0.3 \le p_g < 0.7 \\ \exp loitation & , p_g \ge 0.7 \end{cases} \tag{26}$$

The parameter ranges for each phase are specified in Table 4.

Table 4. Parameter range for each phase.

Phase	Phase progress	Range crossover rate	Mutation rate
Exploration	[0,0.30)	[0.80,0.95]	[0.10,0.30]
Transition	[0.30,0.70)	[0.70,0.85]	[0.08,0.20]
Exploitation	[0.70,1.0]	[0.60,0.75]	[0.05,0.10]

(2) Improvement rate calculation

Definition 1 (population diversity measure): Let $\delta(g)$ denote the diversity index of the gth generation population.

Definition 2 (improvement rate): Let $W = 10$ represent the sliding window. The improvement rate is defined as

$$P(g) = \frac{f_{best}(g-W) - f_{best}(g)}{f_{best}(g-W)} \tag{27}$$

where $f_{best}(g)$ denotes the optimal fitness value of the gth generation.

Definition 3 (diversity adjustment factor):

$$\partial_{\text{div}} = \text{II}[\delta(g) \geq \delta_{threshold}] \tag{28}$$

Where $\delta_{threshold} = 0.3$, $\text{II}[\cdot]$ is the indicator function.

Case 1: Fitness improvement ($f_{best}(g) < f_{best}(g-1)$)

$$P_{\text{c}}(g+1) = \begin{cases} \min(p_c(g) + 0.03, p_c^{\max}), \partial_{div} = 1 \\ \min(p_c(g) + 0.02, p_m^{\max}), \partial_{div} = 0 \end{cases} \tag{29}$$

$$P_{\text{m}}(g+1) = \begin{cases} \max(p_m(g) - 0.02, p_m^{\max}), \partial_{div} = 1 \\ \max(p_m(g) + 0.02, p_m^{\max}), \partial_{div} = 0 \end{cases} \tag{30}$$

Case 2: No improvement and stagnation ($f_{best}(g) \geq f_{best}(g-1) \wedge \text{p}(g) < 0.01$)

$$P_c(g+1) = \max(P_c(g) - 0.05, P_c^{\min}) \tag{31}$$

$$P_m(g+1) = \min(P_m(g) - 0.05, P_c^{\max}) \tag{32}$$

4.6. Improved cross-operation

The crossover operation uses the edge recombination crossover method to increase the quality and stability of paths, prevent premature convergence, avoid creating duplicate nodes, promote population variety, and better retain the structural consistency of parent paths. Here, a mutation probability is introduced to encourage the combination of advantageous genes when population diversity is too large. The specific steps are as follows:

Where diversity is the current population's diversity measure, α is the adjustment coefficient, and is the baseline mutation probability. The following are the precise operating steps:

Figure 4 displays the two parent pathways prior to the operation.

Complete parent population 1	**Vehicle route**	0	1	7	8	9	6	10	4	2	3	0
	Drone itinerary	0	0	0	1	0	0	0	1	0	0	0

Figure 4. Two parent paths before the crossover operation.

Edge recombination graph of the parents: 0→1, 1→7, 7→8, 8→9, 9→6, 6→10, 10→4, 4→2, 2→3, 3→0

Edges selected from Parent 1: 0→1, 1→7, 7→8, 8→9

Through edge selection, a new offspring path is generated by sequentially connecting the chosen edges from the parent paths.

Offspring 1 (Figure 5):

0	1	7	8	9	6	10	1	4	2	3	0
0	0	0	0	1	0	0	0	1	0	0	0

Figure 5. Path of offspring 1 generated via edge recombination crossover.

Corrected offspring path 1 (shown in Figure 6):

0	7	8	9	6	10	1	4	2	3	0
0	0	0	1	0	0	0	1	0	0	0

Figure 6. Corrected path of offspring 1.

4.7. Improved mutation operations

Four mutation operations are applied to truck paths, with adaptive transformations based on the algorithm's implementation progress. After a new truck route is generated, the exact dynamic programming procedure described in Section 4.3(2) is invoked to optimize the corresponding drone route and form a complete chromosome.

The specific transformation strategies are as follows:

(1) Adaptive mutation operator selection

Let the set of mutation operators be denoted as $M = \{M_1, M_2, M_3, M_4\}$, with the corresponding operations listed in Table 5.

Table 5. Operations of adaptive variation operators.

Operator	Operation
M1	2-opt
M2	Reverse
M3	Insert
M4	Relocate

(2) Weight update mechanism

Let $w_i^{(t)}$ denote the weight of mutation operator i in generation t, satisfying:

$$\sum_{i=1}^{4} w_i^{(t)} = 1, \forall t \tag{33}$$

Step 1: Calculate the success rate

$$\eta_i^{(t)} = \frac{s_i^{(t)}}{a_i^{(t)}} \tag{34}$$

Where $s_i^{(t)}$ represents the number of successful attempts, and $a_i^{(t)}$ represents the total number of attempts.

Step 2: Calculate the average improvement rate.

$$l_i^{(t)} = \frac{1}{s_i^{(t)}} \sum_{k=1}^{s_i^{(t)}} l_{i,k} \tag{35}$$

Normalization improvement rate:

$$l_i^{(t)} = \frac{l_i^{(t)}}{\max_j l_j^{(t)} + \varepsilon} \tag{36}$$

Step 3: Perform the comprehensive calculation

$$\theta_i^{(t)} = 0.7 \cdot \eta_i^{(t)} + 0.3 l_i^{(t)} \tag{37}$$

Step 4: Weight update (exponential smoothing)

$$\mathrm{w}_i^{(t+1)} = 0.7 \cdot w_i^{(t)} + 0.3 \cdot \frac{\theta_i^{(t)}}{\sum_{j=1}^{4} \theta_i^{(t)}} \tag{38}$$

Step 5: Normalization

$$w_i^{(t+1)} = \frac{w_i^{(t+1)}}{\sum_{j=1}^{4} w_j^{(t+1)}} \tag{39}$$

Taking the reverse as an example, the operational process is illustrated in Figure 7.

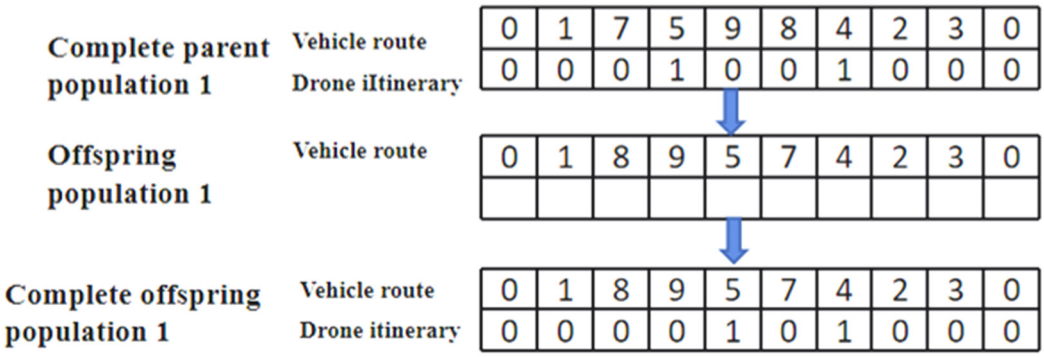

Figure 7. Mutation operation process.

While keeping the truck path codes unchanged, perform a swap mutation operation on the drone paths to generate new drone path codes. This process creates new chromosomes by maintaining the truck path codes while applying the swap mutation to the drone paths. This operation enhances population diversity, enabling newly generated individuals to possess stronger exploration capabilities

within the solution space. Simultaneously, it ensures generated solutions are valid and more optimal, thereby improving the quality of emergency supply distribution plans.

4.8. Termination conditions

To obtain high-quality solutions within reasonable computational time, the HAGA algorithm employs a preset termination criterion based on the maximum iteration generation. When the population evolution reaches the preset maximum generation $G_{\max}$, the algorithm automatically terminates. This mechanism prevents infinite algorithmic runs and ensures satisfactory solutions within finite computational time.

For constraint handling, a sufficiently large penalty factor P is employed. This ensures that infeasible solutions receive significantly lower fitness scores during evaluation, effectively eliminating them during selection. Simultaneously, the magnitude of the penalty is carefully controlled to prevent excessive distortion of the search process due to overly stringent punishment.

5. Case study testing and result analysis

5.1. Data sources

The data sources used were R201, C201, and RC201 from the Solomon case studies as the foundation for large-scale simulations. Smaller-scale simulations were then derived by reducing the number of points in the original case studies [39]. The large-scale simulation defines 100 affected points, the medium-scale simulation defines 50 affected points, and the small-scale simulation defines 25 affected points. The distribution center is designated as node 0. Distances between nodes are calculated using Euclidean distance. Specific parameter settings are detailed in Table 6, with congestion zone configurations based on Liu Changshi's methodology [34].

Table 6. Baseline parameter configuration.

Parameter	Value
Truck speed	50 (km/h)
Truck payload	500 (kg)
Truck service time	90 (s)
Drone speed	50 (km/h)
Drone payload	50 (kg)
Drone flight endurance	25 (km)
Drone service time	60 (s)
Drones per truck	3
Blocked region 1	Center: (10,40), Radius: 6
Blocked region 2	Center: (55,35), Radius: 7
Blocked region 3	Center: (46,10), Radius: 8
Blocked region 4	Center: (60,60), Radius: 9

5.2. Test analysis

The suggested HAGA is implemented in this study using PyCharm Community Edition 2024.1 as the development environment. Three algorithms, GA [34], VNS [33], and HAGA, were used to solve the model and fully assess HAGA's performance. All comparison algorithms (GA, VNS) were run in the same development environment to guarantee fairness. The same termination conditions used in the HAGA method were used with the other algorithms, and preliminary experimental tweaks were made to optimize their parameters (such as population size and number of iterations). The details are as follows:

Based on the standard Solomon test set, nine test cases were selected from [39]: R206, C206, and RC206 at small, medium, and large scales. For small and medium-scale cases, the population sizes were 25 and 50, respectively, with 400 iterations. The program terminated if fitness failed to improve after 80 iterations. For the large-scale test cases, the population size was 100 with 200 iterations; the program terminated upon reaching the iteration limit. To mitigate randomness, each test case underwent 10 independent solves. Among the 10 solution results, the minimum delivery time was recorded as the optimal value; the average of the 10 delivery times served as the solution mean; and the average of the 10 solution execution times represented the average runtime. The following metrics were defined: α denotes the improvement rate of the optimal solution, β represents the improvement rate of the solution mean, and θ indicates the reduction rate of the average runtime.

(1) Case solution on a small scale

Small-scale case studies were solved using the standard Solomon test set comprising RC206(25), R206(25), and C206(25). Specifically, RC206(25) contains 21 truck task points and 4 drone task points; R206(25) contains 21 truck task points and 4 drone task points; and C206(25) includes 20 truck task points and 5 drone task points.

a. Validation of the HAGA algorithm's effectiveness

To validate the effectiveness of the HAGA algorithm, this paper employs the OR-Tools solver to solve small-scale instances C201, R201, and RC201. The optimal solutions obtained from OR-Tools serve as the baseline for comparison with HAGA's results. For HAGA, each instance was run independently five times, recording the optimal value and average value. The comparison results are shown in Table 7.

Table 7. Comparison of OR-Tools and HAGA's solution results.

Data type	OR-Tools (optimal)	HAGA best	Best gap (%)	HAGA mean	Mean gap (%)
C201	473.90	473.90	0.000	473.90	0.002
R201	366.14	366.14	0.000	366.14	0.000
RC201	523.07	523.07	0.000	523.07	0.000

Here, $Gap(\%) = \frac{Bast_{HAGA} - optimum_{OR-Tool}}{optimum_{OR-Tool}} \times 100\%$.

Table 7 shows that OR-Tools obtained optimal objective values of 473.90, 366.14, and 523.07 for C201, R201, and RC201, respectively. HAGA's optimal results for all three instances perfectly matched those of OR-Tools, with corresponding optimal solution deviations of 0.00%. Based on the average results from five runs, the average deviation for R201 and RC201 is 0.00%, while C201 shows an average deviation of only 0.002%. This demonstrates that HAGA achieves results consistent with

or extremely close to those of an exact solver on small-scale problems, thereby validating its computational effectiveness.

b. Comparison of the HAGA, GA, and VNS heuristic algorithms

The HAGA, GA, and VNS algorithms were each applied to solve each case study through 10 independent experiments. To eliminate randomness effects, the minimum delivery duration was ultimately selected as the optimal solution. Detailed results are presented in Figure 8 and Table 8.

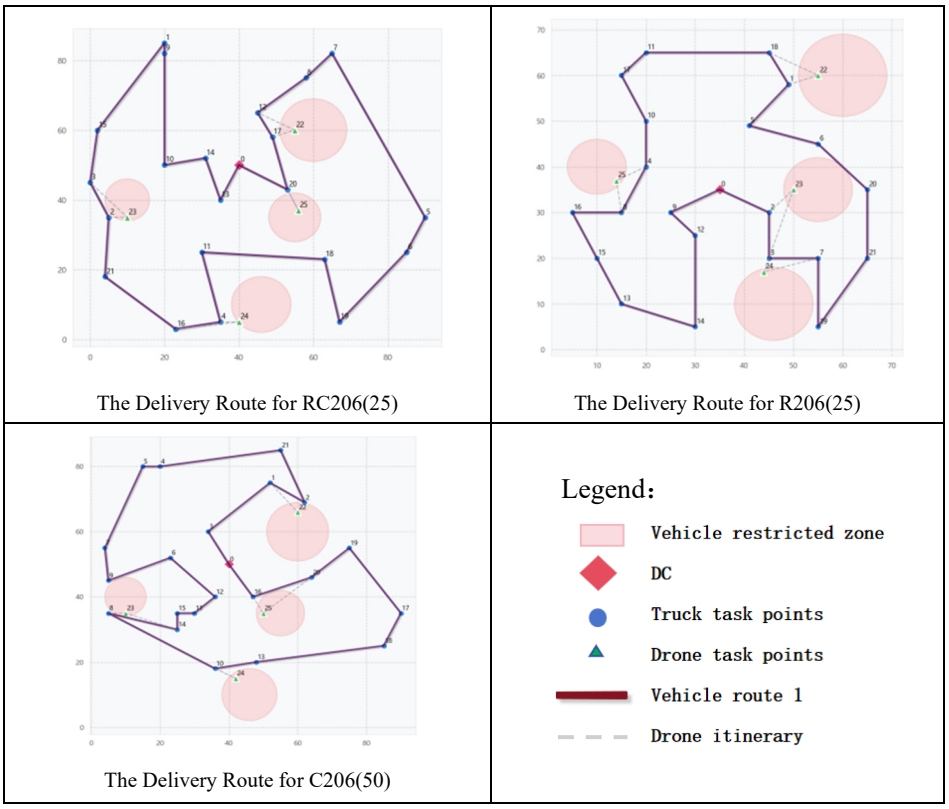

Figure 8. Delivery-route illustration for small-scale instances.

Table 8. Computational results for small-scale instances.

Solving method	Optimum (min)	α (%)	Mean (min)	β (%)	Avg. time (s)	θ (%)	# Trucks
RC201(25)							
HAGA	523.07	-	523.07	-	4.59	-	1
GA	523.07	0	523.07	0	5.43	15.5	1
VNS	523.07	0	523.07	0	5.9	22.2	1
R201(25)							
HAGA	366.14	-	366.14	-	6.01	-	1
GA	366.14	0	366.14	0	7.24	17	1
VNS	366.14	0	366.14	0	8.13	26.1	1
C201(25)							
HAGA	473.9	-	473.91	-	4.09	-	1
GA	473.9	0	473.91	0	5.79	29.4	1
VNS	473.9	0	473.91	0	5.33	23.3	1

Figure 8 and Table 8 present the results for small-scale test cases. The experimental findings indicate the following: (1) For the small-scale RC206, R206, and C206 test cases, the path results obtained by the HAGA, GA, and VNS algorithms are consistent. (2) In the R206(25) case, both the optimal value and the mean solution obtained by HAGA are essentially consistent with the results from the traditional GA algorithm and the VNS algorithm in [33]. (3) The average runtime of the HAGA algorithm is significantly reduced, shortening by 15.5%–29.4% compared to the GA algorithm and by 22.2%–26.1% compared to the VNS algorithm.

(2) Case solution on a medium scale

Computational tests are conducted using medium-scale cases from the standard Solomon benchmark suite: RC206(50), R206(50), and C206(50). In particular, there are 43 truck task points and 7 UAV-served nodes in RC206(50), 43 truck task points and 7 drone task points in R206(50), and 42 truck-served nodes and 8 drone-served nodes in C206(50). The best answer is determined by solving each case independently 10 times and recording the lowest objective value. Table 9 and Figure 9 show the computational results.

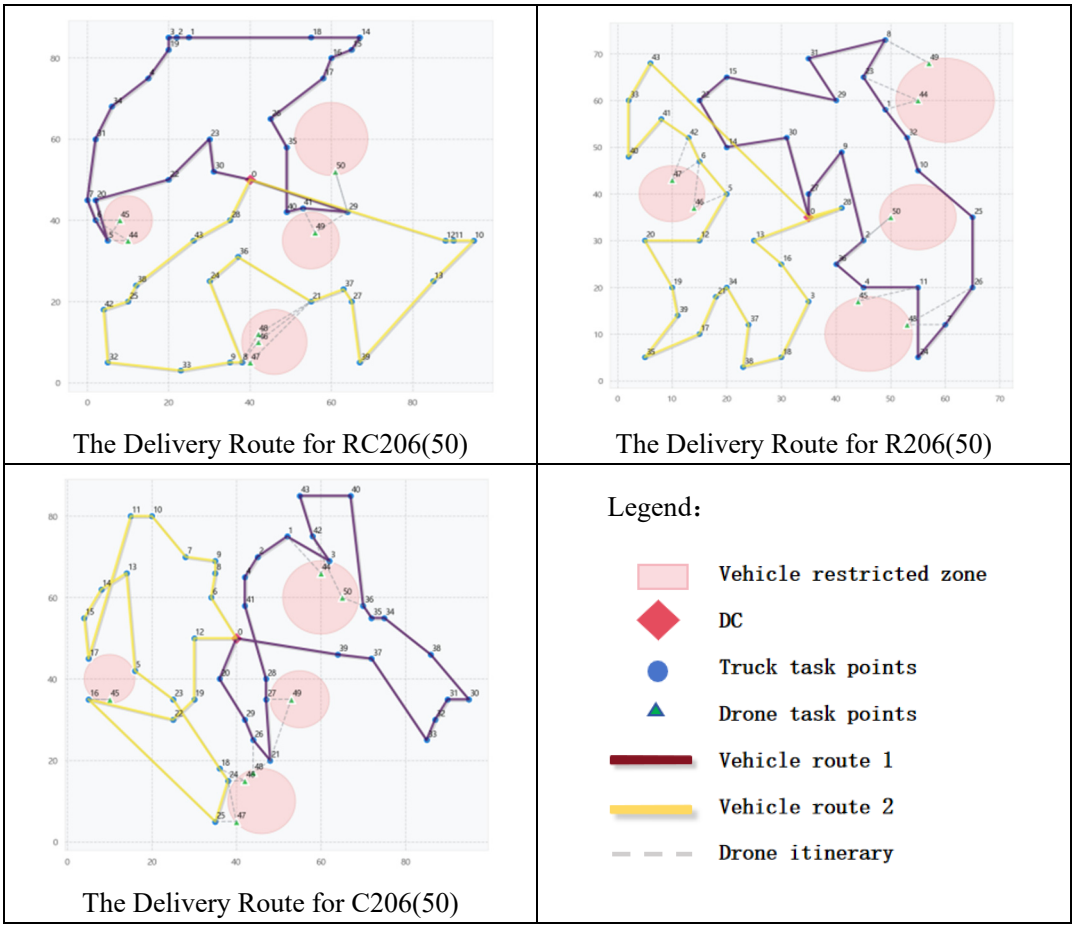

Figure 9. Delivery-route illustration for medium-scale instances.

Table 9. Computational results for medium-scale instances.

Solving method	Optimum (min)	α (%)	Mean (min)	β (%)	Avg. time (s)	θ (%)	# Trucks
RC201 (50)							
HAGA	357.14	-	361.21	-	17.17	-	2
GA	362.73	15.41	408.7	11.62	21.09	18.59	2
VNS	385.63	7.39	406.63	11.17	23.12	25.74	2
R201 (50)							
HAGA	304.87	-	304.87	-	10.05	-	2
GA	304.87	0	304.87	0	15.32	34.4	2
VNS	304.87	0	304.87	0	19.72	49.04	2
C201 (50)							
HAGA	354.07	-	355.95	-	20.26	-	2
GA	361.42	2.03	378.35	5.92	24.01	15.62	2
VNS	388.2	8.79	390.14	8.76	29.24	30.71	2

The following are the results of the medium-scale case studies, as shown in Figure 9 and Table 9: 1) In the RC206(50) and C206(50) cases, HAGA's average delivery time outperformed both the traditional GA algorithm and the VNS algorithm, achieving reductions of 8.77% and 9.97%, respectively; in the RC206(50) and C206(50) cases, HAGA's minimum delivery time was reduced by an average of 8.72% and 8.09%, respectively, in comparison to the GA and VNS algorithms. 2) HAGA's average runtime was 22.87% and 35.16% lower than that of the GA and VNS algorithms, respectively, in the RC206(50), R206(50), and C206(50) test cases. All things considered, the HAGA algorithm successfully increases delivery efficiency and decreases solution time.

(3) Case solution on a large scale

Large-scale case solution testing was performed on RC206(100), R206(100), and C206(100) using the standard Solomon test suite. In particular, there are 81 truck task points and 19 drone task points in the RC206(100) case, 80 truck task points and 20 drone task points in the R206(100) case, and 80 truck task points and 20 drone task points in the C206(100) case. The smallest value was chosen as the best solution after each test case was solved 10 times. Table 10 and Figure 10 present the findings.

The results from the three large-scale test instances in Table 10 and Figure 10 demonstrate that the HAGA algorithm outperforms GA and VNS in both solution quality and computational time for the tested problems. (1) In the RC206, R206, and C206 instances, the optimal solutions obtained by HAGA were 7.90%, 1.02%, and 14.94% better than those obtained by GA, respectively, with an average improvement of 7.95%. Compared to VNS, the corresponding improvements were 2.52%, 1.25%, and 14.38%, respectively, with an average gain of 6.05%. (2) Regarding the average objective value across multiple runs, HAGA outperformed GA and VNS by 7.40% and 9.93%, respectively, indicating superior average performance across these three instances. (3) Regarding computational time, HAGA reduced time by 5.61%, 15.20%, and 15.27% compared to GA, and by 7.69%, 14.46%, and 10.34% compared to VNS. Overall, the computational results for these large-scale emergency-supply allocation instances demonstrate that HAGA is competitive in both solution quality and computational efficiency.

Table 10. Computational results for large-scale instances.

Solving method	Optimum (min)	α (%)	Mean (min)	β (%)	Avg. time (s)	θ (%)	# Trucks
RC201 (100)							
HAGA	252.7	-	261.83	-	120.72	-	4
GA	274.37	7.9	276.18	5.19	127.89	5.61	4
VNS	259.23	2.52	266.25	1.66	130.78	7.69	4
R201 (100)							
HAGA	353.48	-	357.07	-	88.51	-	3
GA	357.13	1.02	379.19	5.83	104.38	15.2	3
VNS	357.97	1.25	362.58	1.52	103.47	14.46	3
C201 (100)							
HAGA	259.82	-	271.3	-	91.98	-	4
GA	305.45	14.94	305.45	11.18	108.56	15.27	4
VNS	303.47	14.38	311.62	12.94	102.59	10.34	4

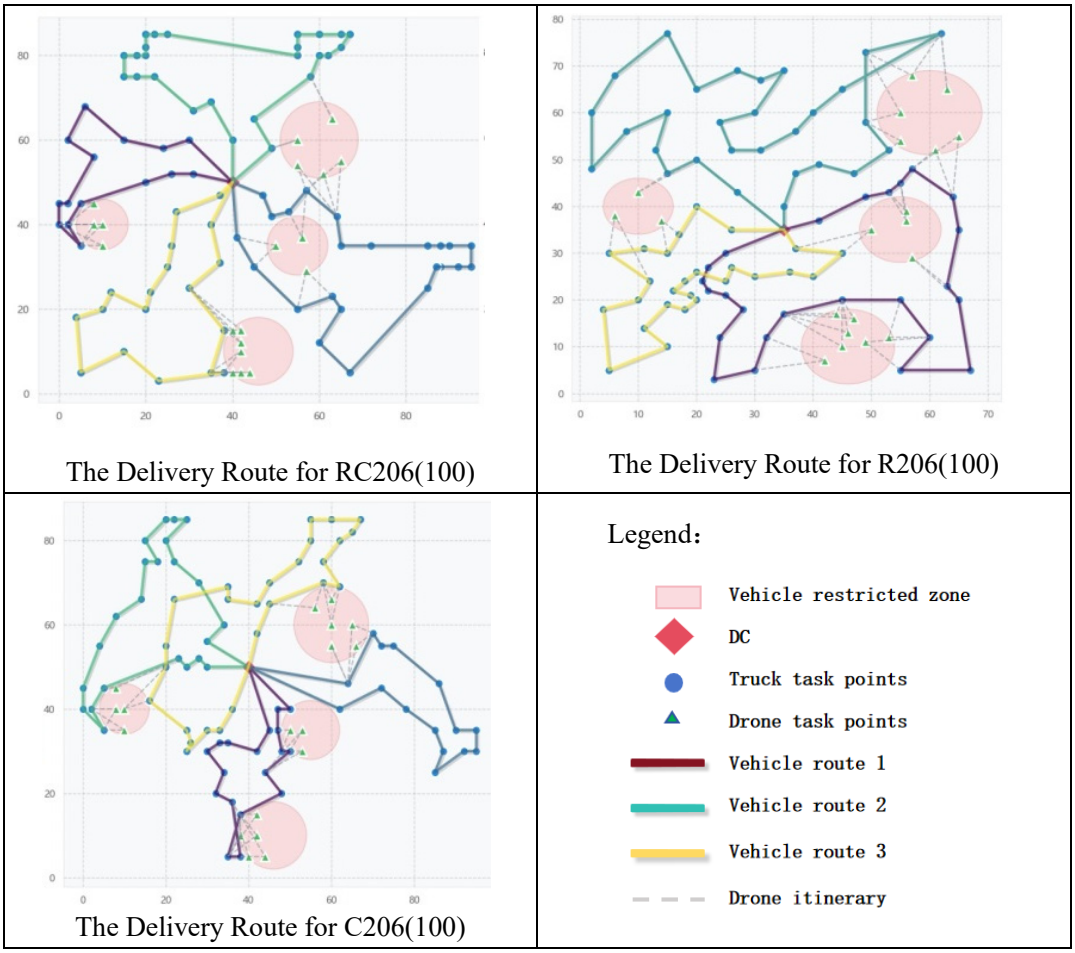

Figure 10. Schematic of delivery routes for large-scale instances.

5.3. Convergence analysis

To examine the convergence performance of the proposed HAGA, this paper selected a large-scale example from RC201. The current optimal objective value changes during the algorithm's iteration process were recorded, and a convergence curve was plotted, as shown in Figure 11. Figure 10 demonstrates that the algorithm continuously improves the objective value within the first 20 generations, exhibiting strong global search capability. Between generations 20 and 45, the objective value decreases rapidly, indicating the algorithm effectively identifies superior solutions and progressively escapes local search regions. Upon reaching the 45th iteration, the algorithm first attained the optimal objective value of 252.7. Subsequently, the curve remained largely stable, indicating that the algorithm had converged. This result demonstrates that HAGA possesses both strong global exploration capabilities and strong local exploitation capabilities in the later stages, enabling it to obtain stable and high-quality solutions within a finite number of iterations.

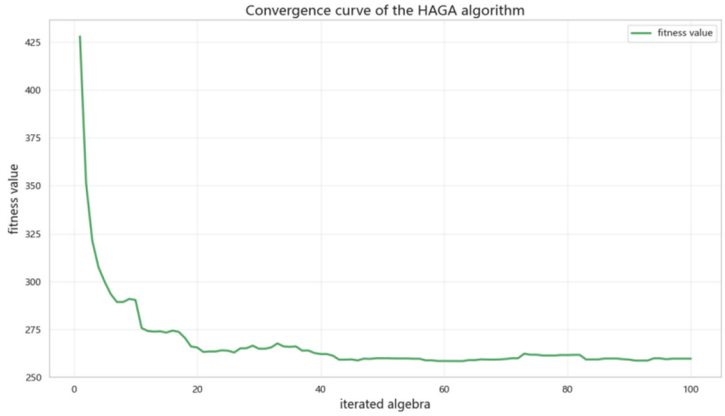

Figure 11. Convergence curve of the objective function value, for the example RC206(100), using the HAGA algorithm.

5.4. Sensitivity analysis

This section primarily conducts sensitivity analysis using the medium-scale case study RC201(50) as the test subject. Since the problem assumes that the drone's payload capacity always meets the requirements at each task point (i.e., demand splitting scenarios are not considered), the sensitivity analysis of the drone's maximum payload capacity is not discussed here. Instead, the focus is on analyzing the sensitivity of the drone's maximum flight range and speed while keeping other parameters constant.

5.4.1. Maximum drone flight range sensitivity analysis

A sensitivity analysis of the maximum drone flying range is conducted under the assumption that other parameters remain constant. The impact of increasing the maximum flight range in 5-km increments from 25 to 60 km on the overall emergency supply delivery time (objective function) is examined. Figure 12 shows the specific results.

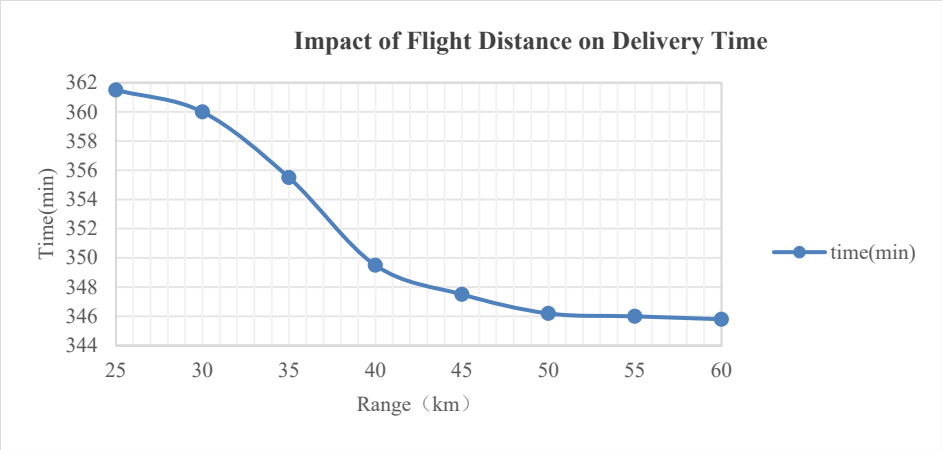

Figure 12. Impact of flight distance on delivery time.

Figure 12 demonstrates that the overall delivery time of the emergency supplies shows a declining trend in tandem with the increase in the drone's maximum flight distance. The objective function value drops significantly as the flight distance increases from 30 to 50 km, indicating that the drone can reach more disaster-stricken locations, reducing the truck–drone synergy time and improving the overall delivery efficiency. However, the rate of decrease of the objective function value slows down as the flight mileage reaches 50 km. This suggests that the marginal benefit of the flight mileage gradually decreases after the number of task points for the drone's single-launch delivery service increases to a certain extent. As a result, the drone's flight mileage in this case study is set at approximately 50 km. This can both fully utilize the advantage of the drone's quick reaction time and prevent resource waste caused by blindly pursuing a longer flight distance.

5.4.2. Drone speed sensitivity analysis

To examine the effects of flying speeds ranging from 30 to 60 km/h on the goal function, the initial drone speed was gradually increased by 5 km/h while keeping all other variables constant. Figure 13 displays the results.

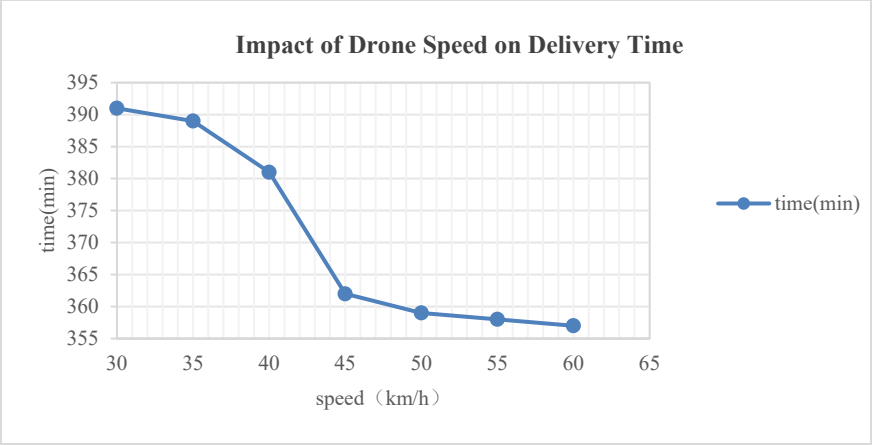

Figure 13. Impact of drone speed on delivery time.

Figure 13 illustrates how the overall time is significantly reduced when the flight speed is increased from 35 to 55 km/h. This shows that the drone can complete the delivery task more rapidly, reducing both the truck waiting time and the overall delivery time. That is, it reflects the improvement in task efficiency brought about by the drone's rapid response capability. The typical law of diminishing marginal returns is evident when the speed is increased to 50 and 60 km/h, even though the delivery time continues to decrease. This indicates that the drone's flight speed in this case may have a crucial turning point at 55 km/h, beyond which the benefits of increasing the unit speed are greatly diminished.

The sensitivity analysis reveals a nonlinear relationship between technical parameters and overall system performance in emergency logistics. Notably, enhancing a single drone capability—such as speed or range—yields diminishing marginal returns. As a result, the study recommends that decision-makers focus on achieving an optimal balance between cost and operational effectiveness when designing emergency logistics systems. Empirical results indicate that setting the drone speed to approximately 55 km/h and the range to around 50 km provides the best system efficiency. This balanced approach helps prevent unnecessary resource expenditure that can occur when pursuing a single high-performance metric to excess.

6. Conclusions

6.1. Summary of key findings

This paper introduces a collaborative path optimization strategy involving multiple trucks and drones to enhance material delivery in emergency logistics scenarios. By integrating detailed modeling, innovative algorithm design, and comprehensive empirical analysis, the proposed approach systematically improves the efficiency and effectiveness of emergency response operations. The following elements provide a summary of the particular research content.

(1) Grounded in the post-disaster scenario, six foundational assumptions are formalized to delineate the model's boundary conditions. We develop a novel multi-truck, multi-drone, multi-loop collaborative routing optimization model for emergency logistics, with the objective of minimizing the makespan. This model theoretically extends the classical vehicle routing problem literature by explicitly incorporating "access restrictions" as a core modeling construct and enabling drone multi-point delivery per flight, thereby providing a more accurate representation of post-disaster logistics realities where infrastructure degradation and timeliness are primary concerns.

(2) To solve the resultant NP-hard problem, the HAGA is devised. Its design incorporates problem-specific strategies such as accessibility-based task partitioning, two-dimensional encoding, and greedy initialization. Crucially, the integration of adaptive genetic operators (with dynamically adjusted crossover and mutation probabilities based on population diversity and individual fitness) with dynamic programming for initial solution generation represents a theoretical contribution to metaheuristic design, demonstrating a principled approach to balancing exploration and exploitation in complex combinatorial landscapes.

(3) The algorithm's effectiveness was systematically evaluated using Solomon benchmark instances at multiple scales. HAGA demonstrated superior performance in both computational efficiency and solution quality when compared to traditional GA and VNS methods. Specifically, HAGA reduced the shortest delivery job completion time by up to 15.41% compared to GA and 14.38%

compared to VNS. For medium- and large-scale applications, HAGA's solution runtime was 34.4% shorter than that of GA and 49.04% shorter than that of VNS.

(4) A systematic sensitivity analysis transcends mere parameter tuning, offering fundamental insights into the performance drivers of collaborative systems. It identifies an optimal efficiency envelope for drone operations (e.g., at 55 km/h speed and 50 km range), thereby enriching the theoretical understanding of the trade-offs between drone capabilities and system-wide logistics performance in time-constrained environments.

6.2. Managerial implications

The practical implications of this study not only support emergency managers but also reinforce its theoretical contributions by demonstrating the real-world validity of the proposed model and algorithm. The findings offer crucial information for the detailed implementation and administration of the logistics scenario in the low-altitude economy:

(1) The creation of a "truck backbone–drone terminal" distribution network maximizes distribution efficiency by optimizing drone parameters and leveraging the complementary strengths of truck bulk transport and drone speed and flexibility. This network design puts into practice the theoretical principles of collaborative optimization under access constraints, illustrating how the extended vehicle routing problem (VRP) framework can effectively guide resource allocation in real-world logistics operations.

(2) Beyond emergency logistics, the research findings are applicable to a range of scenarios requiring rapid response, such as last-mile distribution in e-commerce, the transportation of medical first-aid supplies, and mountain logistics. These applications not only validate the generalizability of the proposed theoretical model but also provide essential methodological support and practical guidance for advancing the industrial application of the low-altitude economy in smart logistics.

In summary, the study bridges the gap between theory and practice, offering actionable insights for both emergency and commercial logistics settings within the evolving landscape of low-altitude airspace utilization.

6.3. Research limitations and future avenues

Although this study has made progress in the field, future research directions have been identified to address the inherent limitations of deterministic modeling and further enhance the theoretical and practical impact of this line of research. Several limitations of this study should be acknowledged. For instance, this model assumes that demand and road conditions are known, with constant travel speeds, and does not account for dynamic traffic conditions or time window issues. Given the high uncertainty in post-disaster environments, future research could incorporate time-varying speed parameters or establish time windows based on real-time disaster information. Additionally, sensitivity analysis of process metrics (such as synchronous waiting ratios and drone utilization rates) indicates that subsequent studies should conduct multi-scenario hierarchical analyses and incorporate operational metrics to thoroughly examine the origins of the algorithm's advantages.

These will also become the core focus of our future research. This will not only address the limitations of this study but also provide policymakers with more robust and adaptive tools, thereby enhancing the resilience and agility of disaster response operations.

Use of Generative-AI tools declaration

No generative AI tools were used in the preparation of this manuscript. All content was created, written, and reviewed by the authors without the assistance of AI tools.

Author contributions

Yubo Sun: Conceptualization and design, research methods, software development, validation, research work, drafting, review, and editing, research implementation. Weihua Liu: Conceptualization and design, supervision, review and editing, project management, fundraising. Jiaqin Hao: Methodology, validation, formal analysis, data organization. Zhentao Shao: Software development, validation, data organization, visualization.

Acknowledgments

The authors are grateful for the financial support provided by the Major Program of the National Social Science Foundation of China (22&ZD139), National Social Science Foundation of China (23BGL146), Scientific Research Project of Higher Education Institutions in Anhui Province (2024AH051811) and Educational Research Project of Xuzhou University of Technology (YGJ2501).

Conflict of interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

1. A. Al-Hilo, M. Samir, C. Assi, S. Sharafeddine, D. Ebrahimi, UAV-assisted content delivery in intelligent transportation systems—joint trajectory planning and cache management, *IEEE Trans. Intell. Transp. Sy*st., **22** (2020), 5155–5167. https://doi.org/10.1109/TITS.2020.3020220
2. I. Dayarian, M. Savelsbergh, J. P. Clarke, Same-day delivery with drone resupply, *Transp. Sci.*, **54** (2020), 229–249. https://doi.org/10.1287/trsc.2019.0944
3. B. D. Song, H. Park, K. Park, Toward flexible and persistent UAV service: Multi-period and multi-objective system design with task assignment for disaster management, *Expert Syst. Appl.*, **206** (2022), 117855. https://doi.org/10.1016/j.eswa.2022.117855
4. C. C. Murray, A. G. Chu, The flying sidekick traveling salesman problem: Optimization of drone-assisted parcel delivery, *Transp. Res. Part C*, **54** (2015), 86–109. https://doi.org/10.1016/j.trc.2015.03.005
5. M. Wohlsen, The next big thing you missed: Amazon's delivery drones could work—they just need trucks, *Wired: Business*, Jun. 2014. Available from: https://www.wired.com/2014/06/amazon-drones/.
6. Clover, Alibaba trials China drone deliveries, *FT*, 2015. Available from: https://www.ft.com/content/3a7f1602-ad0e-11e4-bfcf-00144feab7de.

7. M. Jing, JD.com's drone delivery goes into operation, *China Daily*, 2016. Available from: https://www.chinadaily.com.cn/business/2016-06/08/content_25655486.htm.
8. W. Xu, Meituan's drone food deliveries on the rise, *China Daily*, 2023. Available from: https://www.chinadaily.com.cn/a/202309/19/WS6508e4fca310d2dce4bb66a8.html.
9. Y. Liu, J. M. Shi, Z. H. Luo, X. C. Hu, W. Pedrycz, Z. Liu, Cooperated truck-drone routing with drone energy consumption and time windows, *IEEE Trans. Intell. Transp. Syst.*, **25** (2024), 20390–20404. https://doi.org/10.1109/TITS.2024.3478175
10. S. H. Huang, Y. H. Huang, C. A. Blazquez, A. Carola, Solving the vehicle routing problem with drone for delivery services using an ant colony optimization algorithm, *Adv. Eng. Informatics*, **51** (2022), 101536. https://doi.org/10.1016/j.aei.2022.101536
11. 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
12. P. Keenan, J. Panadero, A. A. Juan, R. Martí, S. McGarraghy, A strategic oscillation simheuristic for the Time Capacitated Arc Routing Problem with stochastic demands, *Comput. Oper. Res.*, **133** (2021), 105337. https://doi.org/10.1016/j.cor.2021.105377
13. L. Amorosi, J. Puerto, C. Valverde, Coordinating drones with mothership vehicles: The mothership and drone routing problem with graphs, *Comput. Oper. Res.*, **136** (2021), 105445. https://doi.org/10.1016/j.cor.2021.105445
14. S. Poikonen, B. Golden, The Mother-ship and Drone Routing Problem, *J. Comput. Publ.*, **32** (2020), 249–262. https://doi.org/10.1287/ijoc.2018.0879
15. X. Liang, J. F. Xiao, Pricing decisions in clothing supply chains based on blockchain and government subsidies, *Chin. J. Syst. Eng.,* **40** (2025), 91–106. https://doi.org/10.13383/j.cnki.jse.2025.01.007
16. X. Xin, S. Wang, T. Zhang, Truck-drone supported humanitarian relief logistics network design: A two-stage distributionally robust optimization approach, *Transp. Res. Part C*, **178** (2025), 105231. https://doi.org/10.1016/j.trc.2025.105231
17. S. Poikonen, B. Golden, Multi-visit drone routing problem, *Comput. Oper. Res.*, **113** (2020), 104802. https://doi.org/10.1016/j.cor.2019.104802
18. S. Nucamendi Guillen, D. Flores Diaz, E. Olivares Benitez, A. Mendoza, A memetic algorithm for the cumulative capacitated vehicle routing problem including priority indexes, *Appl. Sci.,* **10** (2020), 3943. https://doi.org/10.3390/app10113943
19. F. R. Zeng, Z. W. Chen, J. P. Clarke, D. Goldsman, Nested vehicle routing problem: Optimizing drone-truck surveillance operations, *Transp. Res. Part C,* **139** (2022), 103645. https://doi.org/10.1016/j.trc.2022.103645
20. K. Peng, J. X. Du, F. Lu, Q. G. 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
21. Z. H. Luo, M. Poon, Z. Z. Zhang, Z. Liu, A. Lim, The multi-visit traveling salesman problem with multi-drones, *Transp. Res. Part C*, **128** (2021), 103172. https://doi.org/10.1016/j.trc.2021.103172
22. Z. Y. Jin, K. K. H. Ng, C. L. Zhang, W. Liu, F. N. Zhang, G. Y. Xu, A risk-averse distributionally robust optimisation approach for drone-supported relief facility location problem, *Transp. Res. Part E*, **186** (2024), 103538. https://doi.org/10.1016/j.tre.2024.103538

23. A. Amirsahami, F. Barzinpour, M. S. Pishvaee, A fuzzy programming model for decentralization and drone utilization in urban humanitarian relief chains, *Transp. Res. Part E*, **195** (2025), 103949. https://doi.org/10.1016/j.tre.2024.103949
24. J. Duan, H. Luo, G. Wang, Approaches to the truck-drone routing problem: A systematic review, *Swarm Evol. Comput.*, **92** (2025), 101825. https://doi.org/10.1016/j.swevo.2024.101825
25. S. K. Rahimi, D. Rahmani, An improved ALNS for hybrid pickup and drones delivery system in disaster by penalizing deprivation time, *Comput. Oper. Res.*, **170** (2024), 106722. https://doi.org/10.1016/j.cor.2024.106722
26. Y. Shi, J. H. Yang, Q. Han, H. Song, H. X. Guo, Optimal decision-making of post-disaster emergency material scheduling based on helicopter-truck-drone collaboration, *Omega*, **127** (2024), 103104. https://doi.org/10.1016/j.omega.2024.103104
27. L. J. Zhang, Y. Ding, H.Z. Lin, Optimizing synchronized truck-drone delivery with priority in disaster relief, *J. Ind. Manag. Optim.*, **19** (2023), 5143–5162. https://doi.org/10.3934/jimo.2022166
28. T. Mulumba, W. Najy, A. Diabat, The drone-assisted pickup and delivery problem: An adaptive large neighborhood search metaheuristic, *Comput. Oper. Res.*, **161** (2024), 106435. https://doi.org/10.1016/j.cor.2023.106435
29. W. Sun, L. Wu, F. Zhang, Robust optimization for truck-and-drone collaboration with travel time uncertainties, *Transp. Res. Part B*, **204** (2026), 103378. https://doi.org/10.1016/j.trb.2025.103378
30. R. Yan, L. Chen, X. Zhu, Vehicle routing problem with truck and drone under regional restrictions, *Chin. J. Manag. Sci.*, **30** (2022), 144–155. https://doi.org/10.16381/j.cnki.issn1003-207x.2020.0755.
31. K. Y. Zhang, Y. Shi, H. X. Guo, Y. Z. Sun, Optimization Decision-Making for Emergency Material Dispatch in Mountainous Natural Disasters Based on Truck-Drone Collaboration, *Chin. J. Manag. Sci.*, **11** (2024), 1–15. https://doi.org/10.16381/j.cnki.issn1003-207x.2023.1278
32. S. C. Lu, X. L. Shao, H. T. Liu, Y. Wang, Research on Obstacle-Avoiding Delivery of Disaster Relief Materials with Truck–Drone Collaboration, *Comput. Eng. Appl.*, **59** (2023), 289–299.
33. D. W. Hu, S. P. Zhang, H. T. Liu, Y. Wang, Delivery Routing Problem of Truck-Drone Joint Distribution in Initial Stage of Emergency Response, *J. Chang'an Univ. (Nat. Sci. Ed.)*, **44** (2024), 105–119. https://doi.org/10.19721/j.cnki.1671-8879.2024.01.010
34. C. S. Liu, Z. Wu, Z. J. Ma, X. C. Zhou, S. Zhao, P. Sun, Split delivery truck multi-drone collaborative routing for emergency supplies under traffic restriction, *J. Syst. Sci. Math. Sci.*, **43** (2023), 2930–2948.
35. N. Xiao, H. Lan, Two-echelon drone-truck collaborative TSP-based routing for humanitarian logistics with time windows and stochastic demand, *Adv. Prod. Eng. Manag.*, **20** (2025), 351–368. https://doi.org/10.14743/apem2025.3.545
36. J. Q. Zhao, Y. Y. Long, B. L. Xie, G. Y. Xu, Y. W. Liu, A matheuristic solution for efficient scheduling in dynamic truck-drone collaboration, *Expert Syst. Appl.*, **267** (2025), 126218. https://doi.org/10.1016/j.eswa.2024.126218
37. L. Chen, Z. W. Wang, Y. L. Mo, H. O. Pan, Improved genetic algorithms for adaptive replication crossover and mutation, *Comput. Simul.*, **39** (2022), 323–326.

38. M. Srinivas, L. M. Patnaik, Adaptive probabilities of crossover and mutation in genetic algorithms, in Genetic Algorithms: A Case Study in Optimization and Machine Learning, *IEEE Press*, 1994, 123–145.
39. M. M. Solomon, Algorithms for the vehicle routing and scheduling problems with time window constraints, *Oper. Res.*, **35** (1987), 254–265. https://doi.org/10.1287/opre.35.2.254

AIMS Press