


FAIR UAV EMERGENCY SUPPLY DEPLOYMENT BASED ON AN IMPROVED GENETIC ALGORITHM

WEI HONG^{1,2,*} , SHUAICHEN LIU¹, PENGFEI DA¹, WENGJING YIN¹,
XUJIN PU¹ AND SHULING XU¹

Abstract. Unmanned aerial vehicles (UAVs or drones) are widely used in postdisaster material deployment as professional equipment for emergency relief systems. When emergency supplies are deployed to disaster-affected sites, not only the cost and efficiency of rescue should be considered, but also the fairness of material deployment should be given attention. Based on the consideration of fairness, this study investigates the problem of drone emergency supply deployment with dynamic energy consumption constraints. Relative deprivation cost is also used to measure the difference in the psychological trauma of victims at each disaster-affected site to ensure the fairness of supply deployment. A mix integer programming (MIP) model is established, and an improved genetic algorithm, HRGA, is designed to minimize the cost of drone distribution. A heuristic rule is proposed to find an appropriate initial solution quickly by combining the characteristics of the model-related constraints. Compared with the accurate solution of small-scale studies obtained by CPLEX, HRGA reduces the solution time by 68.76% on average while ensuring the accuracy. Compared with the traditional GA, HRGA has certain advantages in solving quality compared with traditional GA with a maximum improvement of 11.55%, and the experimental results verify the feasibility and effectiveness of HRGA. Moreover, the research results can provide useful references for building a fair and efficient drone emergency rescue system.

Mathematics Subject Classification. 90B06, 90C11.

Received March 22, 2023. Accepted August 30, 2025.

1. INTRODUCTION

After the occurrence of a natural disaster, such as earthquakes and floods, air transport is often a direct and effective way of reaching disaster-affected sites, given that ground transportation, such as trucks, cannot reach these sites because of the obstruction of traffic. In China, the 13th Five-Year Plan for the Construction of the National Emergency Response System proposes to use drones as professional equipment for the emergency response system [12], with decision makers relying on ground control systems to monitor drone flights, plan routes for them, and achieve precise positioning in a small area to deliver emergency supplies to disaster victims [34,35]. Deploying emergency supplies in disaster areas is vital in humanitarian relief, and the psychological trauma of victims in the face of disaster events while waiting for supplies to be rationed cannot be ignored [18]. A fair

Keywords. Emergency supply, genetic algorithm, fairness, drone.

¹ School of Business, Jiangnan University, Wuxi 214122, China

² Institute for Food Safety Risk Management, Jiangnan University, Wuxi 214122, China

*Corresponding author: hongwei@jiangnan.edu.cn

and effective decision to ration supplies has a positive effect on relieving the suffering of victims [16]. The fair allocation of postdisaster emergency supplies via drones can effectively solve the problem of the “last mile” in humanitarian relief.

The payload limitation of drones must be considered for emergency material distribution tasks. Murray *et al.* [30] studied the problem of collaborative distribution between a single drone and a single truck. On the basis of the traveling salesman problem (TSP) model, they constructed the flying sidekick TSP (FSTSP) model and the parallel drone scheduling TSP (PDSTSP) model. The load level of the drone plays a vital role in both models. Wang *et al.* [44] expanded the model based on Murry’s models. The path planning model of multi-drone and multi-truck collaborative distribution was constructed as the vehicle routing problem with drone (VRPD) model. Given a specific number of drones and drone speed, they investigated the critical impact of drone load capacity on flight path decision-making. Given that drones are usually battery powered, the endurance limitations of drones should be considered when tasked with emergency material delivery. Dorling *et al.* [8] proposed the first experimental energy-payload consumption estimation model for drones. This model links the dynamic load data of drones to endurance time. The impact of factors, such as drone battery weight, load weight, and number of drones, on total delivery cost and total delivery time was investigated. Jeong *et al.* [22] proposed the FSTSP-ECNZ (FSTSP that implements energy consumption and no-fly zone) model based on Dorling’s experimental approach to investigate further the effect of no-fly zones on drone delivery paths while considering the role of drone weight on energy consumption.

In the early stage of natural disaster events, the demand for emergency supplies increases dramatically at each disaster-affected site. Thus, the psychological trauma of victims increases in varying degrees during the emergency supply delivery via drones. Holguín-Veras *et al.* [17] systematically studied the relative deprivation theory in humanitarian relief. They pointed out that the relative deprivation of victims at each disaster-affected site should be considered in the decision of emergency supply deployment. Moreover, the cost of the trauma of victims at each disaster-affected site due to untimely receipt of emergency supplies was measured by deprivation cost. Deprivation cost is a measure of the cost of the psychological trauma suffered by the victims at each disaster-affected site due to untimely receipt of emergency supplies. Gutjahr *et al.* [15] further found that pursuing the minimization of deprivation cost in the affected area often leads to inequity. Zhu *et al.* [51] proposed relative deprivation cost to pursue the fairness of emergency supply deployment by measuring the degree of psychological trauma suffered by victims in different disaster areas due to untimely receipt of emergency supplies. Reducing the total relative deprivation costs in the disaster area was also considered to reflect the principle of equity in humanitarian relief.

In summary, the research on the equitable deployment for the emergency supply path planning problem involving drones in distribution tasks is rare at this stage and needs to be studied in depth. Thus, this study investigates the drone routing problem considering equity (DRPE) with drone dynamic energy constraints by combining the energy-payload estimation model of drones. The following are the contributions of the study: (1) a comprehensive consideration of drones carrying multiple emergency supplies at a time to achieve multidisaster distribution and regional material rationing equity based on the dynamic energy consumption constraint of drones; (2) a mixed integer programming model MIP considering equity threshold constraint, energy consumption constraint, and load constraint; and (3) an improved genetic algorithm on a heuristic rule namely, HRGA, proposing a heuristic rule considering the constraints related to the DRPE, thereby enabling the initial solution of good quality to be found at the early stage of algorithm operation. Through a series of example experiments, compared with the accurate solution of small-scale cases obtained by CPLEX, HRGA reduces the solution time by 68.76% on average while ensuring the accuracy, and compared with the traditional GA, HRGA has certain advantages in solving quality compared with traditional GA, and the solution quality is improved with a maximum improvement of 11.55%, and the experimental results verify the feasibility and effectiveness of HRGA.

The remainder of this paper is organized as follows. A review of the related literature is provided in Section 2. Section 3 describes the problem under consideration, including a mixed integer programming formulation. The proposed solution approach is described in Section 4, while numerical analysis of the solution procedure is provided in Section 5. The paper concludes in Section 6 with a summary and opportunities for further research.

2. LITERATURE REVIEW

In this study, we mainly consider two parts. Firstly, we examine the decision objectives for post-disaster emergency logistics routing optimization, with emphasis on fairness. Secondly, we provide an overview of the intelligent algorithm for solving the combined optimization problem.

2.1. Decision objectives for post-disaster emergency logistics routing optimization

The decision objectives of traditional commercial logistics routing optimization generally hinge on reducing financial expenditures or minimizing distribution distance costs, *etc.* However, based on the humanitarianism of postdisaster relief, the routing decision also requires careful consideration of relief efficiency, relief effectiveness, and relief fairness in the related literature [20]. Relief efficiency is a traditional decision objective based mainly on the economic costs of relief. For example, disaster relief entails consideration of resource deployment costs and transportation costs [29, 36], which are easier to understand as relief efficiency is relatively similar to the performance evaluation indicators of general commercial logistics.

Relief effectiveness refers to a significant criterion of the quality or effectiveness of routing optimization, which mainly involves response time, plan feasibility, and the extent of casualty reduction in the disaster-affected sites. For example, Xiang *et al.* [45] proposed reducing the total waiting time of victims and the expected death rate as the benchmarks for effectiveness evaluation to assess the deployment effectiveness of medical vehicle relief routing in suffering areas. Sun *et al.* [40] considered minimizing the severity of both minor and severe injuries following a disaster and minimizing the total system cost of the relief process as common goals for relief decisions. Perez-Rodriguez *et al.* [33] first introduced the concept of “deprivation cost” to quantify the physical injuries and mental trauma people suffer due to a lack of relief supplies or services. In terms of emergency logistics after disaster, the total cost is minimized as the objective function, including logistics cost, deprivation cost, and other costs. Wang *et al.* [43] proposed “deprivation level” based on theory of deprivation cost and obtained data through an empirical study. His study developed a quantitative numerical scale method by transforming the results into a curve function to estimate the deprivation level.

In the postdisaster emergency logistics routing optimization decision, the significance of relief fairness has been increasingly highlighted. To evaluate the fairness of relief decisions, decision-makers should focus on whether emergency resources can be used or allocated in a well-balanced manner during the relief activities so that victims in the suffering area can be treated equally with others [47, 48]. Existing literature takes different quantitative approaches, such as min-max fairness value, numerical proportional fairness, absolute deprivation, and the Gini index to measure equity in different aspects [28, 31]. Balcik *et al.* [3] suggests that minimizing unmet needs is a significant decision objective for post-disaster emergency routing optimization. Najafi *et al.* [32] suggests that supplies satisfaction rate can also be used to measure the equity of the routing decision because it is easy to be quantified. Kilic *et al.* [24] used the treatment priority to reflect the differences in the health status of disaster victims, as the standard of treatment fairness in emergency path decision-making. After reviewing decision objectives for postdisaster emergency logistics routing optimization, this study analyzes the reason of considering fairness factor in constructing the model. Different from traditional commercial logistics, emergency logistics should consider not only economic cost of the relief program, but also fairness about the social factor. In the process of emergency management and rescue, victims’ psychological state has an important impact on the efficiency of rescue operations. Therefore, adding fairness can better depict the realistic problems.

2.2. Intelligent algorithms for solving combinatorial optimization

Drone routing for postdisaster emergency supplies deployment considering fairness is a combinatorial optimization problem based on the traditional Vehicle Routing Problem (VRP). Zhang *et al.* [49], by studying the traditional VRP and its related variants (CVRP, VRPTW, SDVRP, DVRP), summarized two general approaches to solving VRP-like problems: exact algorithm and metaheuristic algorithm. The metaheuristic algorithm includes the genetic algorithm [23], simulated annealing algorithm [21], particle swarm search algorithm [50], ant colony algorithm [10], tabu search algorithm [5], *etc.*

Genetic algorithm has the characteristics of the multidirectional global search and has been widely used. Tasan *et al.* [41] proposed a genetic algorithm to solve the vehicle scheduling distribution for simultaneous pickup and delivery in reverse logistics. Kim *et al.* [25] uses a genetic algorithm to solve the pollution path optimization problem considering the greenhouse gas emissions, which takes into account not only vehicle travel distance but also driving speed, payload, road slope, and driver habits. Ghezavati *et al.* [13] developed the traditional single-objective problem and used the non-dominated sorting genetic algorithm (NSGA-II) to solve the dual-objective routing optimization considering time windows in reverse logistics. Abbasi Tavallali *et al.* [1] proposed a routing optimization problem to minimize the customer order delivery cost, delay cost, storage cost, and cross-docking cost based on the consideration of the cross-docking process when importing perishable products, which was solved by NSGA-II, considering the uncertainty of customer demand for perishable products. Gao *et al.* [11] use multiple vehicles for reverse logistics recovery by considering the diversity of reverse logistics recycling types. Considering reducing fixed costs of transportation, carbon emissions, time windows, time penalty, and the constraints of multiple vehicles, the study adopted an improved genetic algorithm based on the greedy strategy to enhance the advantages of the initial swarm.

Generally, the literature on postdisaster emergency logistics routing optimization tends to be limited to quantifiable numerical indicators or economic cost levels as a measure of fairness. However, the poverty, suffering, and negative emotions of victims in suffering areas have been ignored. The suffering perception of the victims is a key factor affecting the allocation of emergency resources and logistics. The psychological trauma victims suffer while waiting for relief may be a good measure of fairness. Simultaneously, new studies have shown that merely minimizing deprivation costs will lead to great inequality [15]. This body of literature provided a new perspective, and explores the influence of fairness on rescue efficiency when considering the psychological trauma of victims. While widely used, genetic algorithm is prone to fall into local solutions. The main reason is that the initial solution of national genetic algorithm is generated too randomly and cannot be uniformly distributed throughout the space of feasible solutions affecting the global search capability. Thus, this paper studies the deployment of emergency materials using drones, considering fairness with dynamic energy consumption constraints and uses relative deprivation cost to measure the different degrees of the psychological trauma of victims in each disaster site to ensure fairness of material deployment. When constructing the mathematical model, we add the fairness of material distribution, which not only considers the psychology of the victims, but also conforms to the realistic background of emergency rescue. The model is solved by an improved genetic algorithm, so that it can represent the complexity of the problem in a fidelity more in line with the empirical reality. To solve the problem, an improved genetic algorithm based on heuristic rules considering the characteristics of constraints is proposed, which avoids the premature convergence from the aspect of initial solution generation.

3. PROBLEM DESCRIPTION AND MATHEMATICAL MODEL

This study investigates the drone emergency supply distribution problem with dynamic energy consumption constraints by considering the payload level and energy consumption of drones as well as the principle of fairness in deploying emergency supplies. The truck carries all the drones and supplies to the emergency distribution center. Then, the drones carry the emergency supplies to the disaster-affected sites. The schematic diagram of the drone distribution path is shown in Figure 1. For ease of expression, the sets, parameters, and decision variables in the model are defined as shown in Table 1.

The model assumptions are as follows: (1) The drones are small multirotor drones with a limited number, and all drones are homogeneous [8]; (2) The demand for emergency supplies at each disaster-affected site is known and served only once by one drone; (3) The weight of emergency supplies required at each disaster-affected point does not exceed the maximum safe payload of the drone; (4) The drones are launched from and retrieved by the truck. The service is suspended when they return to the truck after completing a delivery; (5) The service time of the drone at the disaster-affected site is not considered. The drone flies at a stable speed, with a presumed fixed fluid density of air and gravity; (6) The victims of each disaster site will be psychologically traumatized by injustice, and the psychological trauma can be quantified through economic losses [33].

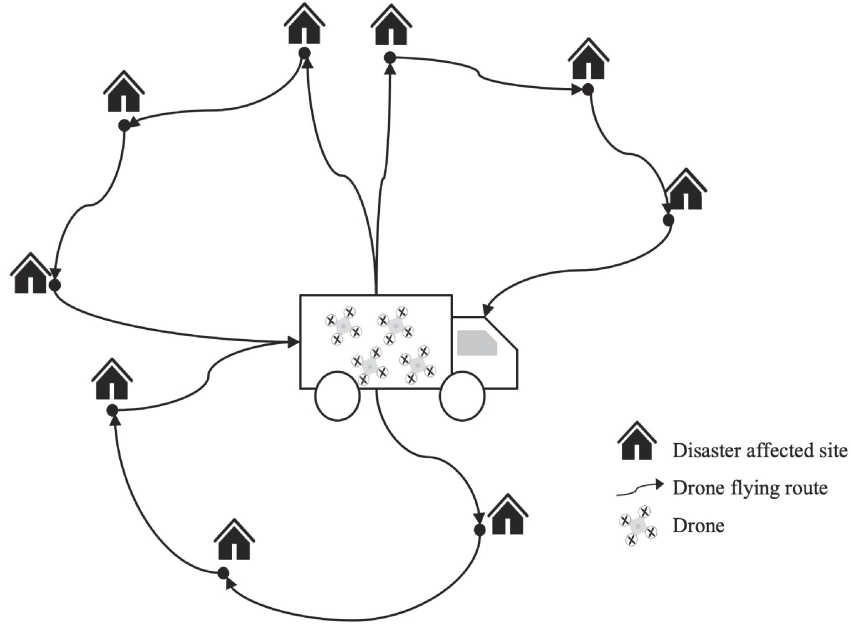


FIGURE 1. Example diagram of drone delivery.

TABLE 1. Notations for the mathematical model.

Variable	Description
N	Set of disaster-affected sites $N = \{1, 2, \dots, n\}$
N'	Set of drones' start and end sites $N' = \{O, O'\}$
V	Disaster-affected sites and the truck stop $V = N \cup N'$
K	Set of drones $K = \{1, 2, \dots, u\}$
D	Set of emergency supplies $D = \{1, 2, \dots, s\}$
g_d	Unit weight of emergency supply $d, \forall d \in D$
Q_{id}	Demand of emergency supply d at a disaster-affected site $i, \forall i \in N, d \in D$
q_{id}	Distribution of emergency supply d to a disaster-affected site $i, q_{id} \leq Q_{id}, \forall i \in N, d \in D$
w_i	Total weight of emergency supplies distributed at a disaster-affected site $i, w_i = \sum_{d \in D} g_d q_{id}, \forall i \in N$
W	Maximum safe payload of the drone
C	Unit path cost of the drone
L	Launch cost incurred after every drone is employed
R	Receiving cost incurred after every drone is employed
E	Battery capacity of the drone
η	Battery energy consumption parameter of the drone
M	An extremely large positive number
u_{ik}	Auxiliary decision variables, avoiding drone sub-loop, $\forall i \in V, k \in K$
w_{ijk}	Payload of drone k on the way from i to $j, \forall i, j \in V, k \in K$
t_{ij}	Time required for a drone flying from i to $j, \forall i, j \in V$
b_{ik}	Remaining battery capacity when drone k arrives at $i, \forall i \in V, k \in K$
s_{ik}	Point of time when drone k arrives at $i, \forall i \in V, k \in K$
x_{ijk}	Binary variable such that $x_{ijk} = 1$ if drone k flies from i to $j, \forall i, j \in V, k \in K; x_{ijk} = 0$, otherwise

Jeong *et al.* [22] assumed that the “energy-payload consumption” estimation model of small multirotor drones is a quasi linear function of power P and load w . On the basis of Jeong’s assumption, we construct a power consumption estimation formula for the drone emergency supply distribution mission in the humanitarian relief scenario. The flight time of the drone between the disaster-affected sites is also introduced. Then, the electrical power e_{ij} consumed by the drone while flying from i to j can be expressed as:

$$e_{ij} = P(w_{ijk})l_{ijk}/v = (\beta_0 + \beta_1 w_{ijk})l_{ijk}/v \quad (3.1)$$

where β_0 and β_1 are the relevant parameters after the linearization of the power function, w_{ijk} is the total weight of the remaining supplies carried by the drone k on the distribution route from i to j , l_{ij} is the distance between i and j , and v is the steady flight speed of the drone.

We referred to the linearization of the deprivation cost function by Biswal *et al.* [4] to approximate the psychological trauma induced by the lack of unit emergency supplies among the victims at each disaster-affected site. Thus, the deprivation cost function DC_i incurred by the disaster-affected site i with the unit penalty cost ω can be expressed as:

$$DC_i = \sum_{d \in D} \sum_{k \in K} [\omega q_{id} s_{ik} + \omega(Q_{id} - q_{id}) * T], \quad \forall i \in N \quad (3.2)$$

where s_{ik} indicates the time point after drone k arrives at the disaster-affected site i , and T is the whole process time for the completion of the distribution task. The cost of deprivation at the disaster-affected sites is composed of two parts: one is the cost of the trauma suffered by the victims from waiting for emergency supplies $\omega q_{id} t_{ik}$, and the other is the cost of the psychological trauma obtained by the victims who have not received the required emergency supplies after the completion of the whole distribution process $\omega(Q_{id} - q_{id}) * T$. Relative deprivation cost is proposed to reflect the principle of fairness in humanitarian relief by measuring the degree of psychological trauma suffered by victims at different disaster-affected sites due to untimely receipt of emergency supplies. This psychological trauma degree is used as an indicator to measure the fairness difference of distribution schemes. The relative deprivation cost RDC_i incurred by the disaster-affected site i can be expressed as

$$RDC_i = DC_i - \min(DC_i), \quad \forall i \in N. \quad (3.3)$$

Therefore, the total relative deprivation costs incurred at all affected sites at the end of the distribution should be maintained within an appropriate upper limit ξ (ξ is a real number) to ensure that the regional distribution scheme for drones is sufficiently equitable. The fairness threshold ξ constrains the feasibility of the solution. Thus, we combine the model of postdisaster humanitarian logistics inventory distribution considering deprivation costs proposed by Perez-Rodriguez *et al.* [33]. Then, the feasible fairness thresholds ξ of the upper and lower bounds can be expressed as:

$$\xi_{\max} = \sum_{i \in N} \sum_{d \in D} \omega Q_{id} T - n * \min(\omega Q_{id} T) \quad (3.4)$$

$$\xi_{\min} = \sum_{i \in N} \sum_{d \in D} \omega Q_{id} l_{Oi}/v - n * \min(\omega Q_{id} l_{Oi}/v). \quad (3.5)$$

If $\xi < \xi_{\min}$, then the fairness is too strong; if $\xi > \xi_{\max}$, then the fairness is too weak. The mix integer programming model MIP constructed in this study is as follows:

$$\text{Min } C \sum_{i \in V} \sum_{j \in V} \sum_{k \in K} l_{ij} x_{ijk} + L \sum_{j \in N} \sum_{k \in K} x_{Ojk} + R \sum_{j \in N} \sum_{k \in K} x_{iO'k}. \quad (3.6)$$

Equation (3.6) indicates that the objective is to minimize the total cost in the whole system. The total cost includes the drone path cost and the fixed cost comprising the launch and receiving costs. The constraints are formed into different equations.

Equation (3.7) indicates that the total relative deprivation costs of all sites in the affected area are within a set threshold ξ to ensure the fairness of supply deployment; if the total relative deprivation costs generated by the distribution scheme exceeds ξ , the scheme incurs the penalty cost caused by unfairness. Equation (3.8) indicates the relative deprivation cost of each disaster-affected site. Equation (3.9) indicates the deprivation cost of each affected site.

$$\sum_{i \in N} RDC_i \leq \xi \tag{3.7}$$

$$RDC_i = DC_i - \min(DC_i), \quad \forall i \in N \tag{3.8}$$

$$DC_i = \sum_{d \in D} \sum_{k \in K} [\omega q_{id} s_{ik} + \omega(Q_{id} - q_{id}) * T], \quad \forall i \in N. \tag{3.9}$$

Equations (3.10) and (3.11) ensure that each affected site can be rescued by the drone, ruling out the case that the drone starts from a site and returns directly to the same site.

$$\sum_{j \in V} \sum_{k \in K} x_{ijk} = 1, \quad \forall i \in N, i \neq j \tag{3.10}$$

$$\sum_{i \in V} \sum_{k \in K} x_{ijk} = 1, \quad \forall j \in N, i \neq j. \tag{3.11}$$

Equation (3.12) ensures that the drone starts from the truck at the beginning of the distribution and returns to the truck at the end. Equation (3.13) guarantees the standard flow balance of the drone path.

$$\sum_{j \in V} x_{Ojk} = \sum_{i \in V} x_{iO'k}, \quad \forall k \in K \tag{3.12}$$

$$\sum_{j \in V} x_{ijk} - \sum_{j \in V} x_{jik} = 0, \quad \forall i \in N, k \in K. \tag{3.13}$$

Equation (3.14) guarantees that no subloop exists in the drone delivery path. Equation (3.15) represents the residual load after the drone leaves i .

$$u_{jk} \geq u_{ik} + w_j - M(1 - x_{ijk}), \quad \forall i, j \in V, k \in K \tag{3.14}$$

$$w_i \leq u_{ik} \leq W, \quad \forall i \in V, k \in K. \tag{3.15}$$

Equation (3.16) represents the load of drone on the way from i to j . Equation (3.17) ensures that the weight of the material carried by the drone is conserved.

$$w_{ijk} \leq W * \sum_{k \in K} x_{ijk}, \quad \forall i, j \in V, i \neq j \tag{3.16}$$

$$\sum_{j \in V} w_{jik} - \sum_{j \in V} w_{ijk} = w_i, \quad \forall i \in N, k \in K. \tag{3.17}$$

Equation (3.18) represents the remaining power of each drone when it arrives at j . Equation (3.19) ensures that the drones to be used are fully charged at the beginning of the distribution.

$$0 < b_{ik} \leq \eta E, \quad \forall i \in V, k \in K \tag{3.18}$$

$$b_{Ok} = \eta E * \sum_{j \in V} x_{Ojk}, \quad \forall k \in K \tag{3.19}$$

Equations (3.20) and (3.21) represent the energy consumed by the drone flying from i to j .

$$b_{jk} \leq b_{ik} - e_{ij} + \eta E(1 - x_{ijk}), \quad \forall i, j \in V, i \neq j, k \in K \quad (3.20)$$

$$b_{jk} \geq b_{ik} - e_{ij} - \eta E(1 - x_{ijk}), \quad \forall i, j \in V, i \neq j, k \in K. \quad (3.21)$$

Equation (3.22) indicates the start time of medical material distribution by drone. Equation (3.23) indicates that the end time of rescue is the delivery time for the last disaster-affected site that received rescue. Equation (3.24) indicates the time of the drone flying from i to j .

$$s_{Ok} = T_{start}, \quad \forall k \in K \quad (3.22)$$

$$T = \max(s_{ik}), \quad \forall i \in N, k \in K \quad (3.23)$$

$$t_{ij} = (l_{ij}/V) * \sum_{k \in K} x_{ijk}, \quad \forall i, j \in V, i \neq j, k \in K \quad (3.24)$$

Equations (3.25) and (3.26) denote the arrival time of the drone at j .

$$s_{jk} \geq s_{ik} + t_{ij} - M(1 - x_{ijk}), \quad \forall i, j \in V, i \neq j, k \in K \quad (3.25)$$

$$s_{jk} \leq s_{ik} + t_{ij} + M(1 - x_{ijk}), \quad \forall i, j \in V, i \neq j, k \in K. \quad (3.26)$$

Equations (3.27) and (3.28) indicate the range of the decision variables.

$$x_{ijk} \in \{0, 1\}, w_{ijk} \geq 0, t_{ij} \geq 0, \quad \forall i, j \in V, k \in K \quad (3.27)$$

$$u_{ik} \geq 0, b_{ik} > 0, s_{ik} > 0, \quad \forall i \in V, k \in K. \quad (3.28)$$

4. IMPROVED GA: HRGA

The path planning problem considering the capacity constraint of the delivery vehicle is NP-hard [7, 39]. The DRPE in this study considers the drone load capacity constraint, the dynamic energy consumption constraint of the drone, and the complex fairness threshold constraint. Therefore, the DRPE is also NP-hard. The genetic algorithm based on a heuristic rule, namely, HRGA, is designed in this study to solve the DRPE in large-scale situations well.

4.1. Initial solution

The following heuristic rule is designed in HRGA to construct the initial solution with good quality.

Step 1: Randomly generate a TSP path

A TSP path is randomly generated for a single drone without considering the drone load, energy consumption, and fairness threshold ξ . The TSP path traverses all disaster-affected points. Then, Step 2 is executed. The pseudocode for randomly generating a TSP path is shown in Algorithm 1.

Algorithm 1. Function for randomly generating a TSP path.

Input: Disaster Affected Site N ; Algorithm Popsiz pop

Output: Tsp Path; Tsp Path Sol

- 1: $TspPath = N, TspPathSol = \emptyset$ // Initialization parameters
 - 2: **for** i **in** $\text{range}(pop)$ **do**
 - 3: $Seed = \text{int}(\text{random.randint}(0, 10))$
 - 4: $\text{Random.seed}(Seed)$
 - 5: $\text{Random.shuffle}(TspPath)$ // Randomly generate a set of TSP path
 - 6: $TspPathSol.append(TspPath)$ // Generate a set of paths that match the population
 - 7: **end for**
 - 8: **return** $TspPath, TspPathSol$
-

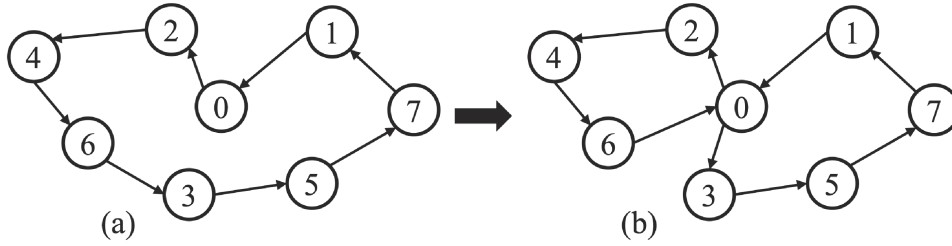


FIGURE 2. Schematic diagram of the split TSP path. (a) Initially randomly generated complete TSP path (0-2-4-6-3-5-7-1-0), where “0” denotes the truck stop and “1-7” denote disaster-affected sites. (b) Split UAV delivery routes (0-2-4-6-0 and 0-3-5-7-1-0), which are split into two feasible routes based on the UAV’s maximum safe payload.

Step 2: Split TSP path

For the TSP path obtained from Step 1, the cumulative weight of the emergency supplies assigned to each disaster-affected site is judged according to the greedy strategy [27] to determine whether the maximum payload capacity of the drone is satisfied in turn. If the weight of the allocated supplies for the next disaster-affected site exceeds the drone’s remaining load capacity, a new path is split from the current site to increase the number of drones enabled; if the generated paths meet the payload capacity of the drones and the number does not exceed the truck capacity, Step 3 is executed. At this time, each drone returns to the initial site after completing the allocation of supplies at the last disaster point. If the number of drones required after the completion of the split TSP path exceeds the truck capacity, the operation is terminated in advance, and no solution is output. The pseudocode for splitting TSP path is shown in Algorithm 2. The following example is given to facilitate the understanding of the split TSP path operation. The affected area presumably has seven affected points, labeled 1, 2, 3, 4, 5, 6, and 7. The truck stopping point is labeled 0. The randomly generated TSP route is “0-2-4-6-3-5-7-1-0” (Fig. 2a). If the payload capacity of one drone cannot take up the distribution work of seven disaster-affected sites, the route is split into two routes from one path. Moreover, the number of drones enabled is increased to two. The routes obtained are “0-2-4-6-0” and “0-3-5-7-1-0” (Fig. 2b).

Algorithm 2. Function for splitting TSP path.

Input: Tsp Path; Drone Payload W ; Load Matrix q

Output: Drone Routes Num; Drone Routes

```

1: DroneRoutesNum = 1, DroneRoutes =  $\emptyset$ , Route =  $\emptyset$ , RemainedCap =  $W$  // Initialization parameters
2: for  $i$  in TspPath do
3:   if RemainedCap -  $q(i) \geq 0$  then
4:     Route.append( $i$ )
5:     RemainedCap = RemainedCap -  $Q(i)$  // The weight of the material of  $i$  is less than the remaining load of the
       drone, that is, it is delivered by the drone
6:   end if
7:   if RemainedCap -  $q(i) < 0$  then
8:     DroneRoutes.append(Route)
9:     Route =  $i$ 
10:    DroneRoutesNum = DroneRoutesNum + 1 // The weight of the material at  $i$  exceeds the remaining load of
       the drone, that is, a new drone is used for distribution
11:    RemainedCap =  $W - Q(i)$ 
12:  end if
13:  DroneRoutes.append(Route)
14: end for
15: return DroneRoutesNum, DroneRoutes

```

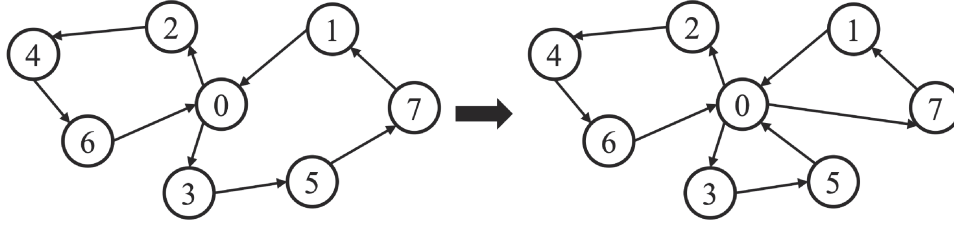


FIGURE 3. Schematic diagram of the adjustment strategy of the execution solution.

Step 3: Calculate the dynamic load of the drone

The dynamic load data of the drone in each route are calculated when it reaches each disaster-affected site according to the set of paths split by Step 2. The weights of the supplies assigned to each disaster point in the path are $[w_1, w_2, w_3, \dots, w_n]$. They are accumulated in reverse order to obtain the cumulative weight of emergency supplies carried by the drone when it arrives at the current disaster-affected site: $[w_n + w_{n-1} + w_{n-2} + \dots + w_1, w_n + w_{n-1} + w_{n-2} + \dots + w_2, \dots, w_n + w_{n-1}, w_n]$. Then, Step 4 is executed. The pseudocode for calculating the dynamic load of the drone is shown in Algorithm 3.

Algorithm 3. Function for calculating the dynamic load of the drone.

Input: Drone Routes; Load Matrix q

Output: All Routes Load

```

1:  $SumRoutesLoad = \emptyset, RoutesLoad = \emptyset$  // Initialization parameters
2: for Route in DroneRoutes do
3:    $DynamicLoad = 0$ 
4:   for  $i$  in  $i, Route$  do
5:      $DynamicLoad += q(i)$  // The dynamic load of the drone at  $i$  is accumulated
6:      $RoutesLoad.append(DynamicLoad)$ 
7:   end for
8: end for
9: return AllRoutesLoad

```

Step 4: Calculate the energy consumption of the drone and adjust the solution

The energy consumption of each drone is calculated from the dynamic cumulative weight sequence obtained by Step 3. If the total energy consumption $\sum_{i \in V} \sum_{j \in V} (\beta_0 + \beta_1 w_{ijk}) l_{ij} / v * x_{ijk}$ does not exceed the maximum releasable energy of each drone battery, the route is established; if the total energy consumption exceeds the maximum releasable energy of each drone battery, the route is not established. According to the “tail customer judgment method” [6, 37], the rejected nodes are placed into a new path to reject the nodes for an unestablished route. The number of drones is increased. Step 3 is performed again, and the operation in a cycle is updated until all the paths meet the power limit. Then, we judge whether the number of drones exceeds the truck capacity. If the number of drones enabled meets the truck capacity, we execute Step 5; if not, we terminate the operation in advance and output no solution. The pseudocode for calculating the energy consumption of the drone and adjusting the solution is shown in Algorithm 4. For example, as shown in Figure 3, the two routes are tested for the drone endurance constraint to understand the operation of adjusting the solution. The energy consumption of route “0-2-4-6-0” is within the drone battery capacity and is retained. The energy consumption of route “0-3-5-7-1-0” exceeds the drone battery capacity. The node at the end of the route is removed and placed into

a new route until the generated new path passes the test of the endurance constraint. The route “0-3-5-7-1-0” generates two new routes, “0-3-5-0” and “0-7-1-0”, after removing nodes.

Algorithm 4. Function for calculating the energy consumption of the drone and adjusting the solution.

Input: Drone Routes; All Routes Load; Distance Matrix L ; Max Battery E

Output: Drone Routes Num, Drone Routes

```

1:  $SumDroneEnergy = \emptyset$ ,  $DroneEnergy = 0$ ,  $NewRoute = \emptyset$  // Initialization parameters
2: for  $Route$  in  $DroneRoutes$  do
3:    $RoutesLoad = AllRoutesLoad(Route)$ 
4:   for  $i$  in  $Route$  do
5:      $DroneEnergy+ = ((\beta_0 + \beta_1 * RoutesLoad(i)) * (L(i, i + 1)v))$ 
6:   end for
7:    $SumDroneEnergy.append(DroneEnergy)$  // Calculate the energy consumption of the drone
8: end for
9: for  $Route, DroneEnergy$  in  $DroneRoutes, SumDroneEnergy$  do
10:  if  $DroneEnergy > E$  then
11:     $NewRoute.append(Route(-1))$ 
12:     $Del(Route(-1))$ 
13:     $DroneRoutes.append(NewRoute)$ 
14:     $DroneRoutesNum = DroneRoutesNum + 1$  // Adjust the solution
15:  end if
16: end for
17: return  $DroneRoutesNum, DroneRoutes$ 

```

Step 5: Determine whether the supply deployment program is fair

The fairness of the scheme is judged by calculating the total value of the relative deprivation cost generated by the material deployment scheme obtained from Step 4. The pseudocode for calculating the total value of relative deprivation cost is shown in Algorithm 5. The default state of relative fairness is reached when the total value of relative deprivation cost is less than the set fairness threshold ξ ; thus, a penalty factor ρ [46] is introduced to make the algorithm actively seek feasible solutions and interfere with the calculation of the deployment scheme cost. The feasible solution with the best possible fairness can be quickly screened to a great extent during the subsequent iterations of GA to obtain a good deployment scheme for the drone material in the

Algorithm 5. Function for calculating total relative deprivation cost.

Input: Drone Routes; Demand Matrix Q ; Load Matrix q ; Time Matrix s ; Punishment Cost ρ ; Equity Value ξ ;

Output: Sum Relative Deprivation Cost RDC ; Sum Punishment Cost PC

```

1:  $RDC = 0$ ,  $PC = 0$  // Initialization parameters
2: for  $Route$  in  $DroneRoutes$  do
3:   for  $i$  in  $Route$  do
4:      $RelativeDeprivationCost = \omega q(i) s(i) + \omega(Q(i) - q(i)) * T$  // Calculate the relative deprivation cost of each disaster-affected site
5:      $RDC+ = RelativeDeprivationCost$  // Calculate the total relative deprivation costs
6:   end for
7: end for
8: if  $RDC > \xi$  then
9:    $PC = \rho * (RDC - \xi)$ 
10:  // Calculate penalty cost
11: end if
12: return  $RDC, PC$ 

```

fair state. The penalty strategy, the most common technique in solving constrained optimization problems [42], uses a penalty factor to deal with the fairness constraint. The solution space can be approximated purposefully during the iterations of GA.

4.2. Fitness function

The drone distribution schemes with sufficient population size are constructed by the heuristic rule. The fitness corresponding to each delivery scheme can be calculated separately. The model's objective function for the DRPE is the minimum problem. Thus, the fitness of each drone delivery scheme Sol in the algorithm is equal to the difference between the maximum of $C(Sol)$ value in all schemes and the current distribution scheme $C(Sol)$, which can be expressed as

$$Fitness(Sol) = \max(C(Sol)) - C(Sol) \quad (4.1)$$

where $C(Sol)$ is the total cost of the distribution scheme $TotalCost$ plus the penalty cost due to noncompliance with the fairness threshold constraint PC . The smaller the value $C(Sol)$ is, the greater the fitness degree of the corresponding scheme is. The $C(Sol)$ is expressed as:

$$C(Sol) = TotalCost + PC. \quad (4.2)$$

4.3. Genetic operator

The real number encoding method is used to encode the drone delivery scheme. The n disaster-affected sites in the whole area are designed as substrings with n genes, randomly arranged from the natural numbers of $1-n$ to represent the random delivery order of the drone to each affected point. For all the affected sites, the distribution scheme is a set of routes encoded by real numbers. All the routes in the set traverse all the affected sites. The schematic diagram of the drone delivery scheme in Figure 3 shows that the number of affected points in the region is 7, that is, $n = 7$. Moreover, the feasible solution shown in the diagram can be expressed as ($|246|35|71|$) by real number encoding. The “|” represents the flight routes of different drones. The truck stop (drone launch point or receiving point) 0 does not need to participate in the real number encoding during the encoding process, which simplifies the encoding step to some extent and does not affect the initial solution result. The specific genetic operation of HRGA is as follows:

- 1) *Selection operator.* The binary tournament selection method is adopted to improve the accuracy of the solution and guarantee the convergence of HRGA as soon as possible by selecting the best individuals in the population into the offspring population. The operation is repeated until the size of the new population reaches the same size as the parent one.
- 2) *Crossover operator and mutation operator.* Order crossover (OX) operator and binary mutation operator are used to perform crossover and mutation operations on individuals in the population. The operation of the OX operator is shown in Figure 4a. Any two chromosomes in the population are selected as a pair of chromosomes involved in the crossover operation, taking ($|246|35|71|$) and ($|246|35|71|$). For example, Figure 4a shows that the fourth to the sixth gene in the chromosome is taken as the fixed gene sequence (as shown in the red numerical part of the figure). The rest of the gene sequence is cleared as the crossover sequence for the complementation operation. The gene complementation is performed according to the gene order of the other chromosome. Thus, the duplicated genes are omitted until all genes are inserted to form the complete chromosomal offspring.

The generated two chromosomal offspring have a certain probability of undergoing a binary mutation operation, as shown in Figure 4b. The third gene of the two chromosomes is mutated and exchanged. Thus, the corresponding sixth gene also undergoes a mutation exchange operation to form a completely mutated chromosomal offspring. This mutation operation changes the traditional mutation method, effectively overcomes premature convergence, and improves the optimization speed of GA.

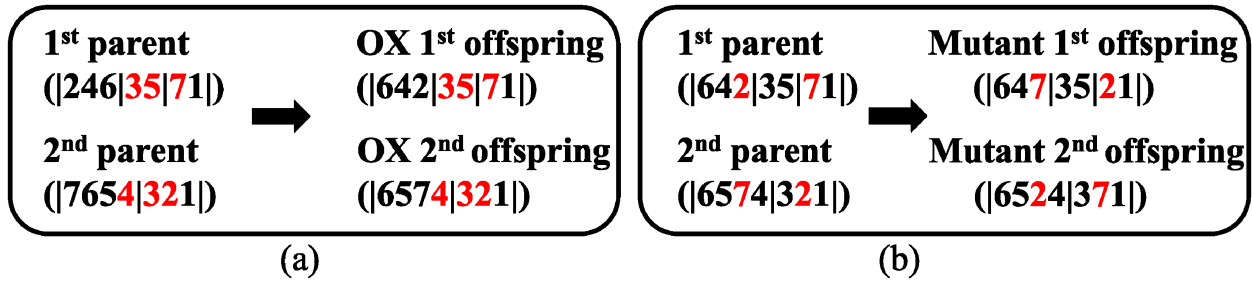


FIGURE 4. Schematic diagram of the OX operator operation and the binary mutation operation.

5. NUMERICAL EXPERIMENTS AND ANALYSIS

A series of arithmetic cases was designed and tested for numerical experiments to verify the effect of HRGA. The results of HRGA were compared with those of the MIP model and traditional GA. HRGA was implemented with Python programming. The compiler is Python 3.8, and the computer configuration is an Intel Core i5-7200U CPU running at 2.50 GHz and 4 GB. The MIP model was implemented by CPLEX 12.8.

5.1. Cases and parameter settings

The experimental cases were selected from the open-source data of Augerat [2] in the CVRP standard case library. They were improved as follows to adapt to the realistic situation of the model. The emergency material type, which was not heterogeneous, was set to 1. The unit weight of the material was set to 0.1kg to adapt to the drone weight. The horizontal and vertical coordinates of the nodes were multiplied by 0.5 to be controlled within 50. The unit used was kilometers to meet the realistic situation of the affected sites. The stable flight speed of the drone was set to 10 km/h, and the parameters in the energy consumption function and deprivation cost function of the drone were set to $\beta_0 = 1.58$ [51], $\beta_1 = 0.217$ [51], and $\omega = 100$ [4].

Algorithm parameters have a certain impact on the performance of the algorithm, and excellent parameters can improve the performance of the algorithm to a certain extent, so it is very important to determine the reasonable parameter value. Taguchi design is a parametric design method that originated in the field of quality management and aims to determine the optimal horizontal combination of parameters with the least number of tests, and has been widely used for its superior reliability, reproducibility, and simple analysis [38]. The Taguchi method mainly measures the advantages and disadvantages of parameters through the signal-to-noise ratio, and the larger the signal-to-noise ratio, the more reasonable the corresponding parameter setting. The objective function of the DRPE problem is to minimize the system cost of drone delivery, and the corresponding signal-to-noise ratio formula is selected as follows:

$$S/N = -10 * \log_{10} \left(\sum_{t \in n} \left(\frac{F_t^2}{n} \right) \right) \tag{5.1}$$

where n is the number of executions of the algorithm at each parameter level, and F_t is the response value, that is, the objective function value of the t th experiment. The parameters of traditional GA and HRGA mainly include: population size, crossover ratio, and mutation ratio. The optimization was carried out by the Taguchi analysis method to determine its optimal value. Considering the optimization of orthogonal experimental design, four levels were taken for each parameter, the population size was taken from $\{100, 200, 300, 400\}$, the crossover rate was taken from $\{0.5, 0.6, 0.7, 0.8\}$, and the mutation rate was taken from $\{0.1, 0.2, 0.3, 0.4\}$. The L16 orthogonal test table was used for the analysis, and the experiment was repeated five times for each horizontal combination. Tables 2 and 3 show the orthogonal test results of the Taguchi analysis, the experimental values of the total cost under each scheme, and the corresponding signal-to-noise ratio and mean value.

TABLE 2. HRGA Taguchi design analysis.

No.	Population size	Crossover rate	Mutation rate	Run-1	Run-2	Run-3	Run-4	Run-5	Avg.	S/N
1	100	0.5	0.1	412.17	412.41	401.91	405.83	423.06	411.08	-52.280
2	100	0.6	0.2	345.30	344.20	336.44	338.98	358.51	344.69	-50.751
3	100	0.7	0.3	365.89	365.20	356.60	359.56	404.61	370.37	-51.382
4	100	0.8	0.4	365.59	364.90	356.31	359.27	424.24	374.06	-51.479
5	200	0.5	0.2	307.91	306.07	299.83	301.60	307.91	304.67	-49.677
6	200	0.6	0.1	343.31	342.18	334.49	336.99	343.31	340.06	-50.632
7	200	0.7	0.4	387.25	386.99	377.51	380.92	387.25	383.98	-51.687
8	200	0.8	0.3	311.08	309.30	302.93	304.77	311.08	307.83	-49.767
9	300	0.5	0.3	300.44	298.45	292.51	293.13	375.34	311.98	-49.927
10	300	0.6	0.4	363.56	362.83	354.32	356.24	355.01	358.39	-51.088
11	300	0.7	0.1	317.61	315.96	309.32	310.30	391.88	329.02	-50.384
12	300	0.8	0.2	324.55	346.08	327.04	309.22	289.90	319.36	-50.101
13	400	0.5	0.4	360.05	359.25	350.88	352.72	313.58	347.29	-50.825
14	400	0.6	0.3	352.48	351.53	343.47	345.16	350.72	348.67	-50.849
15	400	0.7	0.2	315.95	314.27	307.70	308.64	293.57	308.02	-49.775
16	400	0.8	0.1	349.65	348.65	340.70	342.33	305.63	337.39	-50.573

TABLE 3. GA Taguchi design analysis.

No.	Population size	Crossover rate	Mutation rate	Run-1	Run-2	Run-3	Run-4	Run-5	Avg.	S/N
1	100	0.5	0.1	434.21	432.89	435.18	434.09	419.34	431.14	-52.693
2	100	0.6	0.2	427.02	410.56	413.14	396.91	373.55	404.24	-52.141
3	100	0.7	0.3	435.74	434.45	436.67	435.62	435.74	435.64	-52.783
4	100	0.8	0.4	466.95	466.29	478.24	466.82	481.57	471.97	-53.479
5	200	0.5	0.2	320.52	346.93	303.86	310.44	320.52	320.45	-50.124
6	200	0.6	0.1	324.67	313.16	327.91	324.58	324.67	323.00	-50.185
7	200	0.7	0.4	451.66	454.89	442.06	437.54	452.34	447.70	-53.021
8	200	0.8	0.3	389.22	397.00	381.12	389.12	432.64	397.82	-52.003
9	300	0.5	0.3	335.11	341.81	328.13	335.02	318.03	331.62	-50.415
10	300	0.6	0.4	448.03	456.99	438.71	447.91	420.75	442.48	-52.921
11	300	0.7	0.1	319.32	325.70	312.67	319.23	327.25	320.83	-50.127
12	300	0.8	0.2	337.04	343.78	330.03	336.95	431.54	355.87	-51.075
13	400	0.5	0.4	428.29	436.86	419.38	428.18	432.06	428.96	-52.649
14	400	0.6	0.3	419.68	418.07	421.95	419.57	383.61	412.58	-52.315
15	400	0.7	0.2	324.55	341.04	310.80	321.47	366.77	332.93	-50.462
16	400	0.8	0.1	369.42	370.81	355.74	369.32	310.74	355.21	-51.028

The following figures show the main effect plots of the average signal-to-noise ratio and the mean main effect plots of each parameter of the two algorithms at different levels. As can be seen in Figure 5, the population size of the HRGA is 200, the crossover ratio is 0.8, and the mutation ratio is 0.2, and the corresponding signal-to-noise ratio is the largest, and the corresponding mean of the total cost is correspondingly smallest, as shown in Figure 6. Similarly, the population size of the GA is 300, the crossover ratio is 0.5, and the mutation ratio is 0.2, and the corresponding signal-to-noise ratio is the largest, and the corresponding mean of the total cost is also correspondingly smallest, as shown in Figures 7 and 8.

Based on the above analysis, the parameters of the algorithms are set as shown in Table 4.

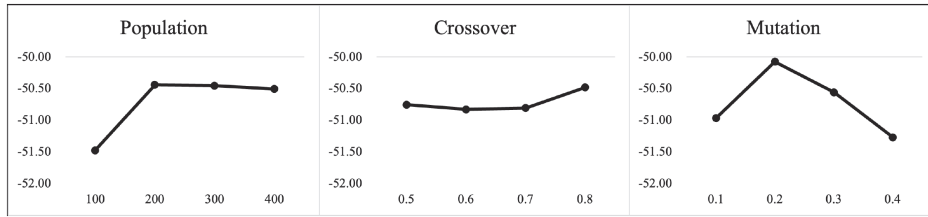


FIGURE 5. Main effect plot of signal-to-noise ratio of HRGA orthogonal test.

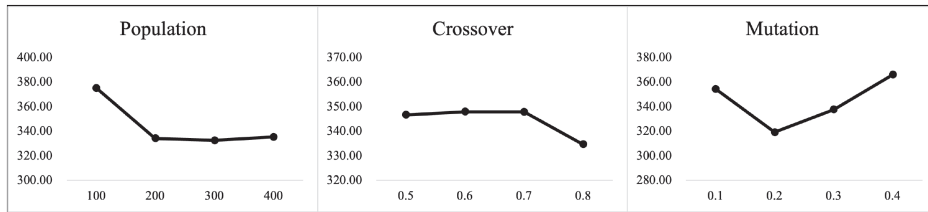


FIGURE 6. HRGA orthogonal test mean principal effect plot.

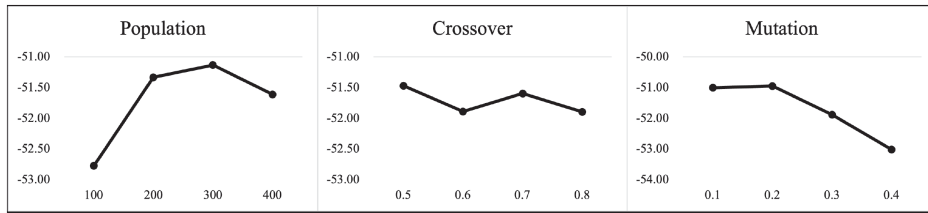


FIGURE 7. Main effect plot of signal-to-noise ratio of HRGA orthogonal test.

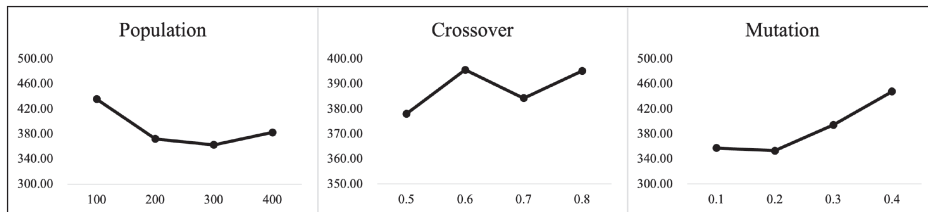


FIGURE 8. GA orthogonal test mean principal effect plot.

5.2. Comparing experiments for solving CPLEX and HRGA

In this paper, CPLEX is used to obtain the exact solution of small-scale cases, and the effectiveness and accuracy of HRGA are compared and verified. The experimental results of small-scale cases are shown in Table 5. In the experimental results, A stands for the abbreviation of Augerat, n denotes the number of affected points, k denotes the number of available drones, RN denotes the number of path strips, TC denotes the total cost, RT denotes the running time, and $Gap_{CPLEX} = (TC(HRGA) - TC(CPLEX))/TC(CPLEX) * 100\%$. The value with an asterisk means that the optimal solution is not obtained by CPLEX in the specified time, and only a feasible solution can be obtained.

TABLE 4. Parameter settings for GA and HRGA.

Algorithm	Parameter	Parameter range	Final parameter value
HRGA	Population size	{100, 200, 300, 400}	200
	Crossover rate	{0.5, 0.6, 0.7, 0.8}	0.8
	Mutation rate	{0.1, 0.2, 0.3, 0.4}	0.2
GA	Population size	{100, 200, 300, 400}	300
	Crossover rate	{0.5, 0.6, 0.7, 0.8}	0.5
	Mutation rate	{0.1, 0.2, 0.3, 0.4}	0.2

TABLE 5. Comparison results of HRGA and CPLEX for solving small-scale cases.

Case name	CPLEX Best Sol.			HRGA Best Sol.			Gap_{CPLEX}	Reduction rate
	RN	TC	RT	RN	TC	RT		
A-n8-k5	3	142.01	3.88	3	144.56	8.31	1.77%	–
A-n9-k5	3	146.83	1.37	3	147.29	8.94	0.31%	–
A-n10-k5	3	147.49	2.57	3	151.20	9.10	2.45%	–
A-n11-k5	3	152.00	3.30	3	153.65	9.61	1.08%	–
A-n12-k5	3	155.47	11.38	3	155.90	10.23	0.27%	10.12%
A-n13-k5	3	156.47	16.28	3	159.15	11.00	1.68%	32.42%
A-n14-k5	3	161.90	41.59	3	164.48	12.18	1.57%	70.71%
A-n15-k5	4	199.29	59.10	4	205.31	17.05	2.93%	71.16%
A-n16-k5	4	202.32	52.74	4	208.84	17.88	3.12%	66.10%
A-n17-k5	4	206.42*	14400	4	209.67	18.57	1.55%	99.87%
A-n18-k5	4	210.49*	14400	4	214.66	18.25	1.94%	99.87%
A-n19-k5	4	220.24*	14400	4	224.59	24.35	1.94%	99.83%
Average	–	175.07	3616.02	–	178.27	13.79	1.72%	68.76%

Notes. These values represent feasible solutions output by CPLEX, as CPLEX failed to find the optimal solution within the specified time (14400 s, *i.e.*, 4 h). All other values without asterisks are exact optimal solutions obtained by CPLEX.

The difference between the solutions obtained by HRGA and the satisfactory solutions obtained by the CPLEX solver is only 0.27%–3.12%. The running time of HRGA is within 25 s, which is a few times less than the computation time consumed by CPLEX. Meanwhile, HRGA corresponds to fewer paths than the solution of CPLEX. This finding indicates that it effectively optimizes the drone number of routes and reduces the drone fixed cost. The DRPE is an NP-hard problem. Thus, the solution time of the CPLEX solver increases exponentially with the increasing size of the cases. Therefore, finding a good solution for large cases is difficult for the CPLEX. HRGA can continuously find good feasible solutions in large population iterations, preserve good solutions by using the binary tournament method, and generate good new solutions with certain probability through selection and variation operators. Thus, the algorithm is prevented from falling into the local optimum quickly. Therefore, HRGA can effectively solve such problems.

5.3. Comparing experiments for solving GA and HRGA

For large-scale cases, this paper compares and analyzes with traditional GA to further verify the advancement and universality of HRGA. For each case, the two algorithms are run independently 20 times under two different fairness thresholds (ξ), ξ_{max} is obtained by equation (3.4), and ξ is set to $0.9\xi_{max}$ and $0.8\xi_{max}$ respectively. The experimental results are shown in Tables 6 and 7, *Best* represents the optimal solution of 20 operations, *Worst* represents the worst solution of 20 operations, *Average* represents the average result of 20 operations,

TABLE 6. Comparison results of GA and HRGA for solving large-scale cases under high fairness threshold.

Case name	$0.9\xi_{max}$	GA					HRGA					Gap_{GA}
		RN	Worst	Best	Average	RDC	RN	Worst	Best	Average	RDC	
A-n30-k8	21497.93	6	389.40	384.54	386.97	18839.91	6	407.42	322.76	373.26	20348.30	-3.54%
A-n32-k8	25430.23	7	488.06	390.86	438.35	22508.96	7	428.79	367.70	403.18	22042.76	-8.02%
A-n34-k8	27992.63	7	487.88	387.93	457.58	23989.56	7	449.59	383.51	422.98	23488.77	-7.56%
A-n40-k10	33943.15	9	620.07	546.04	592.74	29079.67	9	593.96	510.80	551.43	29411.16	-6.97%
A-n42-k10	36377.33	9	634.95	567.94	594.79	32822.85	8	576.65	463.93	527.03	34508.31	-11.39%
A-n45-k10	36725.89	9	730.61	624.94	660.58	34931.51	9	720.07	584.42	618.33	34792.03	-6.40%
A-n50-k12	43053.70	11	865.16	687.14	791.20	39410.47	11	757.47	753.23	754.29	38301.80	-4.67%
A-n55-k12	52084.42	10	800.64	709.31	752.91	49485.82	10	783.65	676.96	703.63	49354.67	-6.54%
A-n70-k16	71220.50	12	1185.43	942.83	1074.78	70877.61	13	1037.39	981.99	995.88	67548.69	-7.34%
A-n74-k16	75058.81	14	1107.79	1007.72	1082.77	70144.13	14	1053.60	1031.67	1042.64	70013.62	-3.71%
A-n90-k18	93539.44	16	1353.30	1254.02	1317.45	88278.22	16	1260.37	1255.08	1258.73	86731.05	-4.46%
A-n95-k18	104718.88	17	1392.79	1347.61	1358.90	95644.28	17	1364.54	1159.21	1273.41	93570.83	-6.29%
Average	-	-	838.01	737.57	792.42	48001.08	-	786.12	707.61	743.73	47509.33	-6.14%

RN represents the average number of drones that are run 20 times, RDC is the average of the total relative deprivation cost of 20 operations, and $Gap_{GA} = (TC(HRGA) - TC(GA))/TC(GA) * 100\%$, when Gap_{GA} is negative, it means that the solution result of HRGA is better than that obtained by GA.

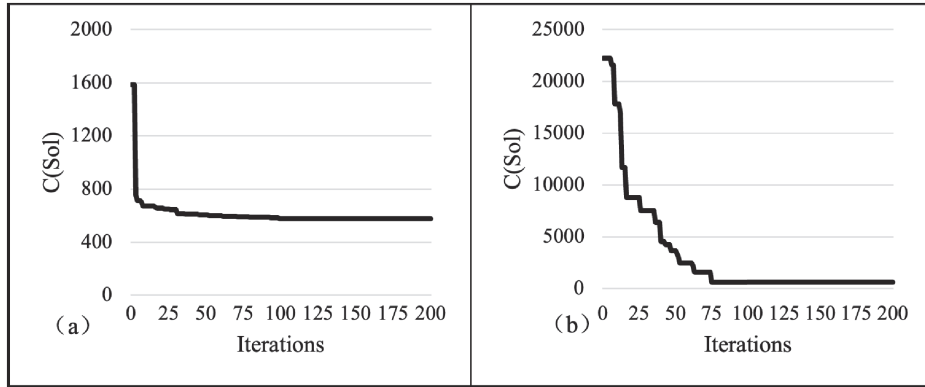
In the case of 30 or more disaster points, the advantage of HRGA is considerable, with a solution quality improvement of 0.68%–11.55%. The satisfactory solutions obtained by HRGA running in all cases are generally better than those of GA. The RDC of the solutions of both algorithms are not greater than the fairness threshold based on the total value of relative deprivation cost set by the model operation. However, the RDC in the optimal solutions obtained by HRGA are generally lower than those obtained by GA because HRGA guarantees the lowest total system cost with the highest fairness under the condition of a given fairness threshold. During the operation of the algorithm, the total relative deprivation costs were reduced by increasing the drone path and reducing the disaster points on each route to ensure the fairness of the distribution process. The solutions corresponding to the total value of relative deprivation cost obtained by HRGA do not exceed the given fairness threshold when the fairness threshold ξ based on the total value of relative deprivation cost is gradually reduced. This finding fully illustrates the feasibility of the algorithm, as shown in Tables 6 and 7. For each case, the TC increases with the decreasing ξ when the drone battery capacities are the same. This scenario verifies that the fairness threshold based on the total value of relative deprivation cost has a constraining effect on the solutions obtained by the algorithm.

5.4. Convergence analysis

Taking the case “A-n45-k10” as an example, the efficiency of the heuristic rule in HRGA for generating initial solutions proposed in this paper is tested. The average optimization total cost $C(Sol)$ using random initial solutions in the GA algorithm is 660.58. The average optimization total cost $C(Sol)$ of HRGA using the heuristic rule is 618.33, and the average running time is 8.4 seconds, which is better than GA. Figure 9a shows the iterative convergence curve when HRGA uses the heuristic rule to construct the initial solution of the drone distribution route, and Figure 9b shows the iterative convergence curve when GA generates the initial solution randomly. HRGA uses a greedy strategy to maximize drone capacity and construct initial solutions, which is different from the random generation method of the original GA algorithm. This greatly saves delivery costs and reduces drone capacity waste, in order to optimize the total cost of the drone allocation system to an approximate optimal solution in the early stages of iteration. Although GA converges quickly, there is still a

TABLE 7. Comparison results of GA and HRGA for solving large-scale cases under low fairness threshold.

Case name	$0.8\xi_{max}$	GA					HRGA					Gap_{GA}
		RN	Worst	Best	Average	RDC	RN	Worst	Best	Average	RDC	
A-n30-k8	19109.27	7	452.75	383.30	411.73	17643.97	6	423.44	374.94	388.13	18430.57	-5.73%
A-n32-k8	22604.65	8	510.93	416.74	480.97	21049.66	7	435.89	417.04	425.43	21281.82	-11.55%
A-n34-k8	24882.34	7	457.97	400.39	441.79	23454.78	7	483.75	417.62	434.15	23378.29	-1.73%
A-n40-k10	30171.69	10	644.50	583.09	613.79	26706.58	9	570.97	566.67	568.82	27486.85	-7.33%
A-n42-k10	32335.40	9	632.94	563.09	615.48	31345.55	9	619.03	588.48	596.12	30479.43	-3.15%
A-n45-k10	32645.24	10	750.32	704.83	716.20	32193.13	10	672.25	655.94	660.02	32443.79	-7.84%
A-n50-k12	38269.96	11	824.12	791.43	815.95	37557.50	11	824.12	785.56	802.96	37987.14	-1.59%
A-n55-k12	46297.26	11	837.46	788.86	801.01	44956.69	11	821.74	786.87	795.59	45288.82	-0.68%
A-n70-k16	63307.11	15	1291.73	945.27	1142.42	62486.54	14	1106.77	955.99	1029.14	61615.56	-9.92%
A-n74-k16	66718.95	14	1176.90	1040.91	1118.01	65623.05	14	1060.50	1040.91	1052.79	63546.57	-5.83%
A-n90-k18	83146.17	16	1368.25	1264.26	1336.70	82204.98	17	1368.82	1217.69	1276.71	81347.71	-4.49%
A-n95-k18	93083.44	18	1448.41	1396.88	1421.92	91399.66	17	1439.87	1229.58	1290.72	90559.85	-9.23%
Average	-	-	866.35	773.25	826.33	44718.51	-	818.93	753.11	776.71	44487.20	-6.00%

FIGURE 9. Convergence curves of $C(Sol)$ function values for HRGA and GA.

gap from HRGA. HRGA quickly optimizes the solution space of the drone's route and speeds up the algorithm convergence rate by judging the drone capacity as well as the battery for several times at the beginning of the algorithm operation, which is better than GA. GA is too blind searching in each iteration to make any judgment on the impractical solution in solution space, so that the optimization effect is not obvious. The HRGA proposed in this paper can effectively improve the convergence efficiency and effect of the solution.

5.5. Sensitivity analysis

In order to identify the influence of key parameters, sensitivity analysis was performed on the maximum drone payload (*Capacity*), drone battery capacity (*Energy*) and fairness threshold (ξ) based on the total value of relative deprivation cost. Each parameter is set at 6 levels, based on case A-n30-k10, a total of 18 scenarios are generated, and each scenario is run independently for 20 times. The average results are shown in Tables 8–11, in which the changes of the total system cost TC and the total relative deprivation cost RDC with parameter fluctuations are shown in Figure 10.

TABLE 8. Effect of different drone payload on path decision objectives.

No.	Capacity	RN	TC	RDC	Optimal route solution
1	5	9	472.48	16242.15	0-6-7-20-9-1-0, 0-8-17-16-4-0, 0-30-18-27-0, 0-12-26-0, 0-13-5-0, 0-10-11-19-0, 0-28-2-14-0, 0-15-23-3-0, 0-21-22-25-24-29-0
2	6	8	435.91	16667.55	0-30-20-11-19-0, 0-26-24-29-9-1-0, 0-22-23-25-0, 0-4-16-17-18-0, 0-12-28-0, 0-5-10-27-0, 0-3-21-13-6-0, 0-2-7-8-14-15-0
3	7	6	345.74	18968.92	0-11-18-5-6-27-0, 0-13-15-14-16-0, 0-4-17-8-7-19-10-0, 0-30-20-9-29-24-1-0, 0-28-2-12-26-21-0, 0-3-25-23-22-0
4	8	6	335.23	20046.07	0-16-14-15-13-0, 0-10-11-19-7-8-17-4-0, 0-18-22-23-21-0, 0-3-25-24-29-9-20-30-0, 0-28-26-2-5-6-0, 0-12-1-27-0
5	9	5	297.41	18261.60	0-8-19-11-10-7-18-0, 0-13-14-4-5-0, 0-21-23-22-15-16-17-6-0, 0-1-30-20-9-2-27-0, 0-28-12-26-3-25-24-29-0
6	10	5	302.13	18275.8	0-6-5-4-16-14-15-0, 0-13-2-1-30-20-9-29-24-0, 0-21-12-10-11-19-27-0, 0-28-26-3-25-23-22-0, 0-7-8-17-18-0

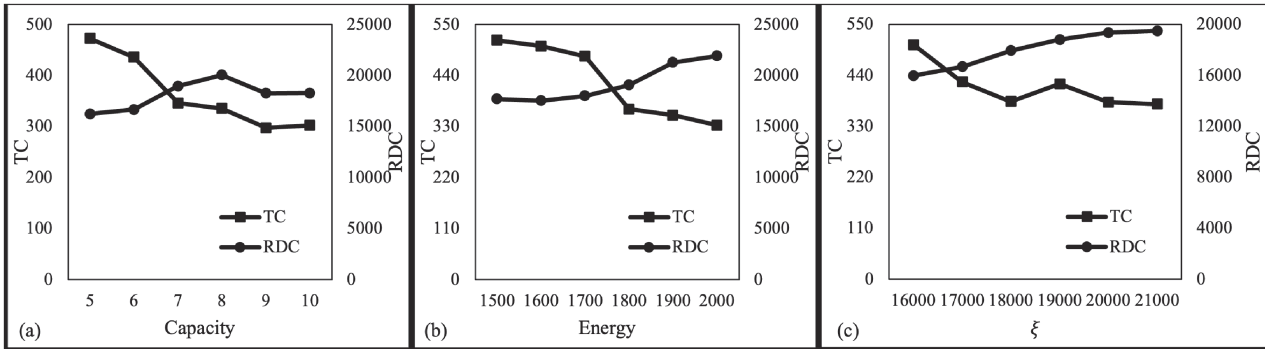


FIGURE 10. The impact of the parameters on TC and RDC .

Change the value of the maximum payload $Capacity$ of the drone and observe the change of the target value of the decision, and the results are shown in Table 8. With the increase of the maximum payload of the drone, the total cost of the system TC will be reduced as shown in Figure 10a, and even the reduction of the number of RNs enabled by the drone, which indicates that the increase of the maximum payload of the drone will affect the reduction of the total cost of the system TC , and the improvement of the payload of the drone is an important factor to reduce the total distribution cost of the drone. When the maximum payload level of the drone gradually increases, the drone will tend to rescue more disaster points if the fairness threshold based on the relative deprivation of the total cost value allows, which may lead to a decrease in the number of drones used (see Figs. 11a and 11b), so the overall cost of the system is reduced.

The change in the target value of the decision was observed by changing $Energy$ of the drone battery capacity. Table 9 shows that the total system cost TC increases with the decrease in the $Energy$ of the drone battery capacity. The increase in the RN is also affected. This finding indicates that reducing the $Energy$ of drone

TABLE 9. Effect of different drone battery capacities on path decision objectives.

No.	Energy	RN	TC	RDC	Optimal route solution
1	1500	9	515.90	17705.90	0-4-16-17-8-7-0, 0-22-15-14-0, 0-2-5-6-13-10-0, 0-28-9-30-1-12-0, 0-27-25-24-26-0, 0-23-21-0, 0-19-29-0, 0-20-3-0, 0-11-18-0
2	1600	9	503.36	17530.12	0-30-24-29-12-0, 0-27-7-19-11-10-20-0, 0-21-15-22-23-0, 0-28-16-6-4-2-0, 0-5-14-8-1-0, 0-3-13-0, 0-9-25-0, 0-18-17-0, 0-26-0
3	1700	8	481.32	18008.23	0-10-19-13-0, 0-4-14-16-18-2-0, 0-5-26-29-24-12-6-0, 0-28-15-3-21-0, 0-27-1-20-9-7-8-0, 0-25-17-0, 0-11-30-0, 0-22-23-0
4	1800	6	367.33	19058.70	0-26-24-29-23-0, 0-28-5-6-4-21-3-25-0, 0-12-13-14-16-17-0, 0-30-20-9-1-10-27-0, 0-18-8-19-11-7-2-0, 0-15-22-0
5	1900	6	354.53	21289.66	0-20-30-11-10-1-0, 0-5-8-18-19-7-28-0, 0-4-17-16-14-13-27-0, 0-6-15-22-23-3-21-0, 0-26-2-12-25-24-29-0, 0-9-0
6	2000	5	332.74	21911.52	0-27-28-21-1-30-10-0, 0-14-15-5-6-18-8-0, 0-13-2-16-4-7-17-0, 0-23-22-3-25-24-26-0, 0-12-29-9-20-11-19-0

battery capacity affects the increase in the TC of the total system cost as shown in Figure 10(b). Moreover, the improvement of the drone endurance capacity is an important factor in the reduction of the total delivery cost of drones. When the drone battery capacity decreases, the drone tends to rescue a few disaster-affected sites. This scenario is allowed by the fairness threshold ξ . Moreover, the implementation of material rationing is accelerated to some extent, and the trauma of disaster victims can be alleviated. The deployment path of each drone is shortened, leading to an increase in the number of drones to be used and the emergence of a single drone dedicated to a single disaster-affected site (see Figs. 11c and 11d). The total cost of the system as a whole increases because each drone has a short deployment route.

The change in the decision target value is observed by changing the value of the fairness threshold ξ . Table 11 shows that the total system cost TC increases gradually with the gradual decrease in the RDC . This scenario also leads to an increase in the number of drones to be used (see Figs. 11e and 11f). If the number of drones to be used increases, the number of disaster-affected sites where each drone is responsible for distribution decreases. This scenario can lead all sites to receive supplies ahead of time to a certain extent. Thus, the victims suffering trauma weaken, and the rescue fairness is high as shown in Figure 10c. This finding suggests that reduced RDC s affect the time of receiving emergency supplies for all affected sites, thereby increasing the total cost of the system. Therefore, the fairness threshold based on the total relative deprivation costs is an important factor for measuring regional equitable rescue. The early time point of receiving emergency supplies at the affected site means that the victims at the affected site do not have to tolerate a long waiting time to receive emergency medical supplies. Thus, each drone may tend to rescue fewer affected sites, depending on the battery capacity. Therefore, the number of drones to be used increases, and the time for emergency medical supply deployment to and from the truck stop is saved. In summary, based on the total value of relative deprivation cost, the impact of drone battery capacity and the fairness threshold must be considered in drone emergency supply deployment path planning.

The sensitivity analysis indicates that the number of drones and the total cost of the system can be controlled well if the decision to deploy emergency supplies is based on the minimum economic cost. However, good results may not necessarily be achieved in terms of relieving the psychological trauma of the victims, which is important

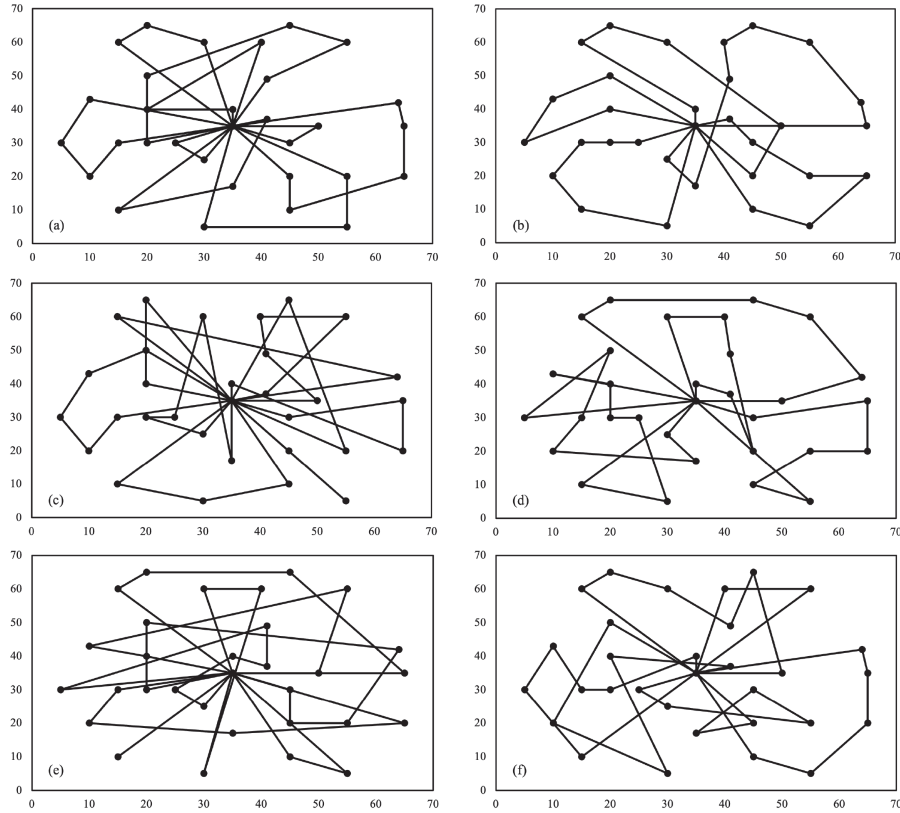


FIGURE 11. Some best solutions by HRGA for A-n30-k10 in different scenarios

TABLE 10. Effect of different fairness thresholds on path decision objectives.

No.	ξ	RN	TC	RDC	Optimal route solution
1	16000	8	506.02	15971.80	0-13-5-27-28-1-17-0, 0-12-9-8-18-0, 0-4-16-2-25-0, 0-26-21-3-29-7-6-0, 0-19-11-20-24-0, 0-10-30-15-0, 0-23-22-0, 0-14-0
2	17000	7	425.60	16680.26	0-5-13-2-28-26-21-0, 0-27-4-16-14-18-7-0, 0-3-23-22-17-6-0, 0-12-1-30-20-15-0, 0-11-19-8-10-0, 0-9-29-24-0, 0-25-0
3	18000	6	383.57	17934.78	0-27-13-2-28-18-7-17-0, 0-1-9-8-4-0, 0-5-16-14-15-25-24-0, 0-26-21-22-3-29-12-6-0, 0-30-20-11-19-10-0, 0-23-0
4	19000	6	421.62	18808.16	0-13-5-16-4-15-0, 0-28-21-3-9-7-18-0, 0-23-25-24-29-6-1-0, 0-26-12-2-22-8-0, 0-27-30-11-10-20-17-0, 0-14-19-0
5	20000	6	382.12	19348.84	0-1-9-8-4-0, 0-28-12-3-25-2-5-18-0, 0-13-6-17-16-14-15-21-0, 0-26-29-30-10-7-0, 0-27-19-11-20-24-0, 0-22-23-0
6	21000	6	377.70	19483.68	0-5-13-3-26-2-21-0, 0-28-18-15-16-7-0, 0-22-23-25-29-24-0, 0-12-20-1-10-11-19-0, 0-27-6-4-8-17-14-0, 0-9-30-0

FUNDING

This work was supported by the National Natural Science Foundation of China (72271109).

DATA AVAILABILITY STATEMENT

The research data associated with this article are included in the article.

REFERENCES

- [1] P. Abbasi Tavallali, M.R. Feylizadeh and A. Amindoust, Presenting a mathematical programming model for routing and scheduling of cross-dock and transportation in green reverse logistics network of covid-19 vaccines. *Iran. J. Optim.* **13**, 2 (2011) 123–146.
- [2] Augerat, Capacitated VRP instances, 2021. Available at <https://neo.lcc.uma.es/vrp/vrp-instances/capacitated-vrp-instances..>
- [3] B. Balcik and B.M. Beamon, Facility location in humanitarian relief. *Int. J. Logist.* **11**, 2 (2008) 101–121.
- [4] A.K. Biswal, M. Jenamani and S.K. Kumar, Warehouse efficiency improvement using rfid in a humanitarian supply chain: Implications for indian food security system. *Transp. Res. E Logist. Transp. Rev.* **109** (2018) 205–224.
- [5] J. Cai, Q. Zhu, Q. Lin, Z. Ming and K.C. Tan, Decomposition-based multiobjective evolutionary optimization with tabu search for dynamic pickup and delivery problems. *IEEE Trans. Intel. Transp. Syst.* **25** (2024) 14830–14843.
- [6] Y.S. Chang and H.J. Lee, Optimal delivery routing with wider drone-delivery areas along a shorter truck-route. *Expert Syst. Appl.* **104** (2018) 307–317.
- [7] M. Dorigo, V. Maniezzo and A. Colomi, Ant system: optimization by a colony of cooperating agents. *IEEE Trans. Syst. Man. Cybern. B Cybern.* **26**, 1 (1996) 29–41.
- [8] K. Dorling, J. Heinrichs, G.G. Messier and S. Magierowski, Vehicle routing problems for drone delivery. *IEEE Trans. Syst. Man. Cybern. Syst.* **47**, 1 (2016) 70–85.
- [9] R. Eberhart and J. Kennedy, A new optimizer using particle swarm theory, in *MHS'95. Proceedings of the Sixth International Symposium on Micro Machine and Human Science*, IEEE, 1995, pp. 39–43.
- [10] Y. Gao, H. Wu and W. Wang, A hybrid ant colony optimization with fireworks algorithm to solve capacitated vehicle routing problem. *Appl. Intell.* **53**, 6 (2023) 7326–7342.
- [11] Z. Gao and C. Ye, Reverse logistics vehicle routing optimization problem based on multivehicle recycling. *Mathematical Problems in Engineering* **9** (2021) 5559684.
- [12] General Office of the State Council of the People's Republic of China, The 13th five year plan for the construction of the national emergency response system [EB/OL], 2017. Available at http://www.gov.cn/zhengce/content/2017-07/19/content_5211752.htm.
- [13] V. Ghezavati and M. Beigi, Solving a bi-objective mathematical model for location-routing problem with time windows in multi-echelon reverse logistics using metaheuristic procedure. *J. Ind. Eng. Int.* **12** (2016) 469–483.
- [14] F. Glover, Future paths for integer programming and links to artificial intelligence. *Comput. Oper. Res.* **13**, 5 (1986) 533–549.
- [15] W. J. Gutjahr and S. Fischer, Equity and deprivation costs in humanitarian logistics. *Eur. J. Oper. Res.* **270**, 1 (2018) 185–197.
- [16] W.J. Gutjahr and P.C. Nolz, Multicriteria optimization in humanitarian aid. *Eur. J. Oper. Res.* **252**, 2 (2016) 351–366.
- [17] J. Holguín-Veras, M. Jaller, L. N. Van Wassenhove, N. Pérez and T. Wachtendorf, On the unique features of post-disaster humanitarian logistics. *J. Oper. Manag.* **30**, 7–8 (2012) 494–506.
- [18] J. Holguín-Veras, N. Pérez, M. Jaller, L. N. Van Wassenhove and F. Aros-Vera, On the appropriate objective function for post-disaster humanitarian logistics models. *J. Oper. Manag.* **31**, 5 (2013) 262–280.
- [19] J. H. Holland, Genetic algorithms and the optimal allocation of trials. *SIAM J. Comput.* **2**, 2 (1973) 88–105.
- [20] M. Huang, K. Smilowitz and B. Balcik, Models for relief routing: Equity, efficiency and efficacy. *Transp. Res. E Logist. Transp. Rev.* **48**, 1, SI (2012) 2–18.
- [21] İ. İlhan, An improved simulated annealing algorithm with crossover operator for capacitated vehicle routing problem. *Swarm Evol. Comput.* **64** (2021) 100911.
- [22] H.Y. Jeong, B.D. Song and S. Lee, Truck-drone hybrid delivery routing: Payload-energy dependency and no-fly zones. *Int. J. Prod. Econ.* **214** (2019) 220–233.

- [23] S. Karakatić, Optimizing nonlinear charging times of electric vehicle routing with genetic algorithm. *Expert Syst. Appl.* **164** (2021) 114039.
- [24] A. Kilic, M.C. Dincer and M.A. Gokce, Determining optimal treatment rate after a disaster. *J. Oper. Res. Soc.* **65**, 7 (2014) 1053–1067.
- [25] W. Kim, J.-G. Kang and H. Kim. Pollution routing problem to reverse logistics of disposed food waste. *Int. Inform. Institute Tokyo Inform.* **19**, 3 (2016) 771.
- [26] S. Kirkpatrick, C. D. Gelatt Jr, and M. P. Vecchi, Optimization by simulated annealing. *Science* **220**, 4598 (1983) 671–680.
- [27] W. Li, L. Xia, Y. Huang and S. Mahmoodi, An ant colony optimization algorithm with adaptive greedy strategy to optimize path problems. *J. Ambient Intell. Humaniz. Comput.* **1** (2022) 1–15.
- [28] J.H. McCoy and H.L. Lee, Using fairness models to improve equity in health delivery fleet management. *Prod. Oper. Manag.* **23**, 6 (2014) 965–977.
- [29] A. Moreno, D. Alem, M. Gendreau and P. Munari, The heterogeneous multicrew scheduling and routing problem in road restoration. *Transp. Res. B Methodol.* **141** (2020) 24–58.
- [30] C.C. Murray and 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.
- [31] D.J. Nair, H. Grzybowska, Y. Fu and V.V. Dixit, Scheduling and routing models for food rescue and delivery operations. *Socio-Econ. Plan. Sci.* **63** (2018) 18–32.
- [32] M. Najafi, K. Eshghi and W. Dullaert, A multi-objective robust optimization model for logistics planning in the earthquake response phase. *Transp. Res. E Logist. Transp. Rev.* **49**, 1 (2013) 217–249.
- [33] N. Perez-Rodriguez and J. Holguin-Veras, Inventory-allocation distribution models for postdisaster humanitarian logistics with explicit consideration of deprivation costs. *Transp. Sci.* **50**, 4 (2016) 1261–1285.
- [34] S. Poikonen and J.F. Campbell, Future directions in drone routing research. *Networks* **77**, 1 (2021), 116–126.
- [35] B. Rabta, C. Wankmueller and G. Reiner, A drone fleet model for last-mile distribution in disaster relief operations. *Int. J. Disaster Risk Reduct.* **28** (2018) 107–112.
- [36] O. Rodriguez-Espindola, P. Albores and C. Brewster, Dynamic formulation for humanitarian response operations incorporating multiple organisations. *Int. J. Prod. Econ.* **204** (2018) 83–98.
- [37] M. Salama and S. Srinivas, Joint optimization of customer location clustering and drone-based routing for last-mile deliveries. *Transp. Res. C Emerg. Technol.* **114** (2020) 620–642.
- [38] H. Soleimani, K. Govindan, H. Saghafi and H. Jafari, Fuzzy multi-objective sustainable and green closed-loop supply chain network design. *Comput. Ind. Eng.* **109** (2017) 191–203.
- [39] N. Stojadinović, B. Bošković, D. Trifunović and S. Janković, Train path congestion management: Using hybrid auctions for decentralized railway capacity allocation. *Transp. Res. A Policy Pract.* **129** (2019) 123–139.
- [40] H. Sun, Y. Wang, and Y. Xue, A bi-objective robust optimization model for disaster response planning under uncertainties. *Comput. Ind. Eng.* **155** (2021) 107213.
- [41] A.S. Tasan and M. Gen, A genetic algorithm based approach to vehicle routing problem with simultaneous pick-up and deliveries. *Comput. Ind. Eng.* **62**, 3, SI (2012) 755–761.
- [42] B. Tessema and G. G. Yen, A self adaptive penalty function based algorithm for constrained optimization, in *2006 IEEE International Conference on Evolutionary Computation*, IEEE, 2006, pp. 246–253.
- [43] X. Wang, X. Wang, L. Liang, X. Yue and L. N. Van Wassenhove, Estimation of deprivation level functions using a numerical rating scale. *Prod. Oper. Manag.* **26**, 11 (2017) 2137–2150.
- [44] Z. Wang and J.-B. Sheu, Vehicle routing problem with drones. *Transport. Res. B Methodol.* **122** (2019) 350–364.
- [45] Y. Xiang and J. Zhuang, A medical resource allocation model for serving emergency victims with deteriorating health conditions. *Ann. Oper. Res.* **236**, 1, SI (2016) 177–196.
- [46] Y. Xiao, Q. Zhao, I. Kaku and Y. Xu. Development of a fuel consumption optimization model for the capacitated vehicle routing problem. *Comput. Oper. Res.* **39**, 7 (2012) 1419–1431.
- [47] L. Yu, H. Yang, L. Miao and C. Zhang, Rollout algorithms for resource allocation in humanitarian logistics. *IJSE Trans.* **51**, 8, SI (2019) 887–909.
- [48] L. Yu, C. Zhang, H. Yang and L. Miao, Novel methods for resource allocation in humanitarian logistics considering human suffering. *Comput. Ind. Eng.* **119** (2018) 1–20.
- [49] H. Zhang, H. Ge, J. Yang and Y. Tong, Review of vehicle routing problems: Models, classification and solving algorithms. *Arch. Comput. Methods Eng.* **29**, 1 (2022) 195–221.
- [50] J. Zhang, F. Yang and X. Weng, An evolutionary scatter search particle swarm optimization algorithm for the vehicle routing problem with time windows. *IEEE Access* **6** (2018) 63468–63485.

- [51] L. Zhu, Y. Gong, Y. Xu and J. Gu, Emergency relief routing models for injured victims considering equity and priority. *Ann. Oper. Res.* **283** (2019) 1573–1606.



Please help to maintain this journal in open access!

This journal is currently published in open access under the Subscribe to Open model (S2O). We are thankful to our subscribers and supporters for making it possible to publish this journal in open access in the current year, free of charge for authors and readers.

Check with your library that it subscribes to the journal, or consider making a personal donation to the S2O programme by contacting subscribers@edpsciences.org.

More information, including a list of supporters and financial transparency reports, is available at <https://edpsciences.org/en/subscribe-to-open-s2o>.