

Article

Optimization of Multi-Cycle Distribution of Emergency Perishable Materials Based on a Two-Stage Algorithm Under Demand Fuzzy

Yang Xu ¹, Xiaodong Li ^{2,*} , Kin-Keung Lai ^{3,*}  and Hao Ji ¹

¹ School of Economics and Management, Xi'an Technological University, Xi'an 710021, China; xuyang@chd.edu.cn (Y.X.); wwwjihao_78@163.com (H.J.)

² School of Economics and Management, Xidian University, Xi'an 710071, China

³ International Business School, Shaanxi Normal University, Xi'an 710119, China

* Correspondence: xiaodong_li1996@163.com (X.L.); mskklai@outlook.com (K.-K.L.)

Abstract

Post-disaster emergency perishable material distribution is an essential part of emergency relief, which is of great significance to reducing disaster losses and casualties and improving rescue efficiency. However, in actual rescue, the demand information of disaster sites is often complex to determine, and the demand for emergency perishable materials needs to be clarified. Therefore, the single-cycle distribution makes it difficult to meet the demand for emergency perishable materials at disaster sites. To effectively improve the efficiency of emergency relief, this paper constructs a distribution optimization model with a multi-cycle vehicle path and the dual objectives of minimizing the distribution delay penalty and corruption cost and minimizing the unsatisfied demand. Initially, the fuzzy demand is addressed through the application of triangular fuzzy numbers and the most probable value weighting method. Following this, a two-stage optimization algorithm is devised by integrating the K-means++ algorithm with an enhanced Differential Evolutionary Whale Optimization Algorithm (DE-WOA). This algorithm operates by first clustering the affected points and subsequently solving the multi-objective model, thereby providing a robust methodology and strategic recommendations for the distribution of perishable materials across diverse scenarios. Our study reveals that the multi-objective model developed in this paper exhibits remarkable operability and practicality in the distribution of post-disaster emergency perishable materials, as evidenced by the verification via numerical examples. Through a comparison with the single-stage whale optimization algorithm, it is evident that the enhanced two-stage differential evolutionary whale optimization algorithm not only demonstrates a substantially faster convergence rate and a superior solution quality but also proves to be more suitably adapted to the proposed model. Significantly, the overall optimization performance has been augmented by 33%, thereby providing further substantiation of the efficacy of the designed improved algorithm.



Academic Editor: Arkadiusz Gola

Received: 9 February 2026

Revised: 6 March 2026

Accepted: 30 March 2026

Published: 3 April 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

Keywords: emergency material distribution; multi-cycle; demand fuzzy; perishable materials; K-means++; whale optimization algorithm

1. Introduction

Large-scale sudden-onset natural or artificial disasters frequently occur worldwide every year, posing serious threats and impacts to society, human production, and life (Chang, Tseng, & Chen, 2007) [1]. How to quickly and effectively respond to these unpredictable

emergencies has attracted significant attention from governments and management at all levels, putting higher requirements forward [2]. After a disaster, the scientific and reasonable distribution of emergency relief materials can effectively reduce casualties, improve rescue efficiency, and alleviate the psychological pressure on victims, making it a critical link in emergency relief operations [3–5]. Among these materials, perishable goods (such as fresh food, medical supplies, and blood) occupy an essential position, and a massive influx of such perishable relief supplies is urgently required at disaster sites.

For the Vehicle Routing Problem (VRP), since it was proposed by Dantzig and Ramser [6] in 1959, VRP theory and models have become relatively mature. In the context of emergency distribution, Tan et al. (2021) established mathematical planning models considering the risk of stochastic supply point failure under fuzzy soft time windows and demand constraints [7]. For post-disaster rescue site selection, Gao et al. (2021) established a multi-stage optimization model under demand uncertainty [8]. To minimize distribution time, Zhou et al. (2021) designed a facility siting-distribution model applicable to post-earthquake relief [9]. Addressing the specific challenges of post-disaster scenarios, Niksirat et al. (2024) formulated a fuzzy multi-objective, multi-period integrated routing-scheduling model [10], while Hssini, I. (2025) proposed a learning-based vehicle routing framework specifically for highly perishable disaster blood logistics [11]. For the medical waste collection problem, Lei and Jia (2026) formulated a multi-period time-varying vehicle routing problem subject to time safety constraints and developed the outer approximation method and a hybrid heuristic algorithm to solve the model [12].

Due to the unique nature of perishable materials, their distribution chain is significantly more complex than that of general goods. Zulvia et al. (2020) established a green VRP model for perishable products considering varying travel times across different periods [13]. Qian et al. (2019) developed a robust optimization model to deal with uncertain demand, maximizing product freshness while minimizing total costs and carbon emissions [14]. Similarly, Leng et al. (2020) formulated a cold chain VRPTW model using multi-objective evolutionary algorithms [15], Gmira et al. (2021) designed an improved tabu search algorithm focusing on time-varying speeds [16], and Rahimi et al. (2017) proposed a multi-period bi-objective model maximizing customer satisfaction by reducing late deliveries [17].

Solving these complex VRP variants, which are NP-hard, typically relies on meta-heuristic algorithms. The Whale Optimization Algorithm (WOA), proposed by Mirjalili et al. [18], has been successfully applied to neural network training [19], resource scheduling [20], and path planning [21] due to its few parameters and convenient operation. However, traditional WOA suffers from premature convergence and easily falls into local optima [22]. Consequently, improving WOA by hybridizing it with other algorithms to solve specific routing problems remains an active research area. Recent studies have made significant strides in this direction; for example, Nguyen, V. M. et al. (2024) developed a hybrid WOA for capacity-limited vehicles [23], and Zhang, X. et al. (2024) proposed a multi-objective learning WOA for open routing networks [24]. Despite the vibrant existing literature on general VRP and fresh product logistics, a critical research gap remains in the intersection of emergency logistics and perishable material distribution. Most current studies either focus on the distribution of general non-perishable relief goods or address the cold-chain logistics of perishables in standard commercial environments. Very few studies comprehensively address the post-disaster distribution of emergency perishable materials under the extreme conditions of severe supply shortages, highly degraded road networks, and the necessity for multi-cycle deliveries. Furthermore, when disaster areas are vast and cross-regional, standard single-stage routing algorithms struggle to balance computational efficiency with the fairness of resource allocation.

Moreover, in the immediate aftermath of a sudden disaster, precise demand data is nearly impossible to acquire due to communication disruptions and chaotic on-site situations. Traditional stochastic models rely heavily on historical probability distributions, which are inapplicable for unpredictable catastrophic events. Therefore, this paper incorporates a robust triangular fuzzy approach to rigorously quantify deep demand uncertainty based on expert estimations and limited initial reports, a methodological trend increasingly validated by recent studies such as Cao et al. (2024), who applied hybrid heuristics to fuzzy routing problems [25]. While prior studies predominantly focus on commercial cold-chain logistics [13,14,16], this research constructs a bi-objective optimization model tailored specifically for the critical initial phase of disaster relief. It explicitly couples time-dependent delay penalties with the physical deterioration costs of perishable materials under severe, multi-cycle supply constraints. Ultimately, to overcome the computational bottlenecks of standard monolithic meta-heuristics, an innovative two-stage optimization framework is proposed. In reality, the area where the affected points are located is often cross-regional due to the wide range of disasters. To make the distribution more “fair,” it is necessary to cluster the affected points by region first and then distribute the demand for the affected points [26]. By synergizing K-means++ spatial clustering with a Differential Evolution-enhanced Whale Optimization Algorithm (K-DEWOA), the proposed methodology effectively decomposes a large-scale, cross-regional NP-hard routing problem into computationally tractable, region-specific tasks.

From a practical and managerial perspective, the business significance of this application is highly profound for emergency management agencies and humanitarian logistics operators. By explicitly quantifying the corruption cost of perishable goods and minimizing delay penalties, this model helps decision-makers reduce the financial waste associated with spoiled medical and food supplies. Furthermore, the “clustering first, routing second” strategy ensures a fairer cross-regional allocation of life-saving resources, thereby enhancing the overall operational resilience, optimizing the utilization of limited rescue vehicles, and ultimately providing a cost-effective, scalable decision-support tool for disaster relief commands.

The main contributions of this paper are listed as follows:

- (1) We formulate a novel bi-objective vehicle routing optimization model for the delivery of emergency perishable materials under supply shortages. This model uniquely integrates fuzzy demand, multi-period delivery, and perishability parameters, providing a mathematical foundation that closely mimics realistic post-disaster uncertainties.
- (2) We propose a two-stage solution framework utilizing the K-means++ clustering algorithm. This framework successfully balances the computational complexity of cross-regional organization with the indispensable requirement of distribution equity in disaster zones.
- (3) We design and validate K-DEWOA, a Whale Optimization Algorithm improved by a Differential Evolution mechanism. This algorithmic advancement successfully overcomes the premature convergence of traditional WOA, demonstrating superior global search capability, faster convergence, and exceptional applicability in solving large-scale bi-objective emergency routing problems, as verified through comprehensive numerical comparisons.

The rest of this paper is structured as follows. Section 2 details the model setup. Section 3 details the design of the model solution and algorithm. Section 4 performs numerical simulation and validation analysis. Finally, Section 5 summarizes the research findings and offers managerial implications as well as potential directions for future research.

2. Problem Description and Modeling

2.1. Problem Description

A simple discrete undirected road traffic network is given as the node set for the distribution center of emergency perishable relief materials, the edge set, and the affected point. The demand for emergency perishable materials at the affected points is expressed in triangular fuzzy numbers. It is known that there is an emergency perishable material in the distribution center, and due to the disaster's suddenness, the material is in short supply. The total amount of the material is, and the corruption rate of the emergency perishable material in the distribution process will change with the change in the distribution time, and the corruption rate is. The demand for emergency perishable materials at each disaster site may not be satisfied at once due to ambiguous demand information and limited vehicle capacity. The distribution center now has a vehicle with a maximum loading capacity of available cars, denotes the distance between any two points, indicates the actual driving speed on the disaster after the disaster occurs, depending on the road conditions, represents the average driving speed of the vehicle, the exact time to arrive at the disaster point is, the ideal time is, defined as the delayed penalty cost, depending on the demand and the degree of destruction at the disaster point, the vehicle is now arranged to depart from the distribution center and provide distribution services to Now, vehicles are arranged to start from the distribution center to carry out distribution services to each disaster point, deliver emergency perishable materials to each disaster point and return to the distribution center until the demand of the disaster point is fully satisfied. Firstly, based on the idea of clustering, the K-means++ clustering algorithm is used to perform regional clustering for the affected points. Then, the triangular fuzzy number and the most probable value weighting method are used to deal with the fuzzy demand and the fuzzy corruption rate. The distribution path and the distribution volume of each demand point are decided with the dual objectives of minimizing the distribution delay penalty and corruption cost, as well as reducing the degree of unsatisfied demand in case of a shortage of materials. For the sake of the problem, the problem model is founded on the following assumptions:

- (1) The location and number of demand points are known;
- (2) Vehicles start from the distribution center and finally return to the distribution center;
- (3) The total amount of material in the distribution center is less than the sum of the material demand at each affected point.

2.2. Description of Variables and Symbols

The symbols and parameter variables used in the model are defined in Tables 1 and 2:

Table 1. Symbols and variables in the model.

Symbols	Meaning
P	Distribution cycle collection
V	Set of all nodes
R	Maximum loading capacity of vehicles
$\tilde{D}_{ip}$	Demand for Emergency Perishable Materials at Disaster Site Cycle p
S	Total amount of materials in the distribution center
t_0	Ideal vehicle arrival time
t_j	Actual vehicle arrival time
c	Penalty cost per unit of vehicle delivery time delay
k	Number of vehicles owned by the distribution center
δ	Minimum permissible material spoilage rate during vehicle transportation
ω	Vehicle utilization rate

Table 2. Decision variables.

Variables	Meaning
x_{ijk}^p	1, if the vehicle transports material from point i to point j in period p 0, otherwise
y_{ik}^p	1, if the task at point i is done by vehicle k 0, otherwise
d_{ipk}	Amount of emergency perishable material supplied to disaster site i by vehicle k during period p

2.3. Modeling

Considering the objectives of the problem and all the constraints, the following model is established.

$$\min \sum_{j \in Z} \sum_{p \in P} (t_j - t_0) c_{1j} x_{ijk}^p + c_{2j} d_{ipk} \theta t_j, \text{ when } t_j \leq t_0, \text{ then } t_j - t_0 = 0 \quad (1)$$

$$\min \left\{ 1 - \sum_{p \in P} d_{ipk} (1 - \theta t_j) / \tilde{D}_{ip}, i \in V, k \in K \right\} \quad (2)$$

$$\text{s.t.} \sum_{i \in Z} d_{ipk} y_{ik}^p \leq R, k \in K, p \in P \quad (3)$$

$$\sum_{i \in Z} \sum_{k \in K} d_{ipk} \leq S, p \in P \quad (4)$$

$$0 \leq \sum_{k \in K} d_{ipk} \theta t_j \leq \tilde{D}_{ip}, i \in V, p \in P \quad (5)$$

$$\sum_{i \in S} \sum_{j \in S} x_{ijk}^p \leq |S| - 1, k \in K, p \in P \quad (6)$$

$$\sum_{j \in Z} x_{ojk}^p = \sum_{j \in Z} x_{jok}^p \leq 1, j \in V, k \in K, p \in P \quad (7)$$

$$\sum_{j \in Z, i \neq j} x_{ijk}^p \leq 1, i \in V, k \in K, p \in P \quad (8)$$

$$x_{ijk}^p \leq y_{ik}^p, i, j \in V, k \in K, p \in P \quad (9)$$

$$0 \leq \frac{d_{ipk} \theta t_j}{d_{ipk}} \leq \delta, (i, j) \in V, k \in K, p \in P \quad (10)$$

$$\frac{d_{ipk}}{R_k} \geq \omega, i \in V, k \in K, p \in P \quad (11)$$

$$x_{ijk}^p = 0 \text{ or } 1, y_{ik}^p = 0 \text{ or } 1, (i, j) \in V, p \in P \quad (12)$$

$$d_{ipk} \geq 0 \quad (13)$$

Equation (1) indicates that the distribution delay penalty and corruption cost are minimized; Equation (2) indicates that the degree of unsatisfied demand is minimized; Equation (3) indicates that the distribution volume of each vehicle in each cycle does not exceed the maximum load capacity of the vehicle; Equation (4) indicates that the total volume of each cycle does not exceed the total volume of materials in the distribution center; Equation (5) indicates that it is ensured that the volume of emergency perishable materials distributed to the disaster-stricken point in each cycle does not exceed its demand, and it cannot be negative; Equation (6) represents the breaking of sub-circuits in the distribution process (Toth and Daniele, 2014) [27]; Equation (7) represents the guarantee that after performing the service to the disaster-stricken point in each cycle, it must return to the

distribution center, and its starting point must also be the distribution center; Equation (8) represents the guarantee that a vehicle will not pass through the same path twice or more times in any cycle; Equation (9) represents that a vehicle will only pass through the side it is located in when it serves the disaster-stricken point; Equation (10) represents the emergency perishable materials in the distribution process of the degree of corruption; and Equation (11) represents the vehicle utilization rate of each cycle. Equation (12) represents the integer variable constraint, and Equation (13) represents the non-negative constraint.

$$\text{While } p > 2, \tilde{D}_{ip} = \tilde{D}_{ip} + \left(\tilde{D}_{i(p-1)} - \sum_{k \in K} d_{ik(p-1)} \right).$$

3. Model Solution

3.1. Clarification of Ambiguous Requirements

In this paper, the fuzzy demand is expressed as a triangular fuzzy number:

$$\tilde{D}_{ip} = (q_{1i}, q_{2i}, q_{3i}), 0 < q_{1i} \leq q_{2i} \leq q_{3i} \quad (14)$$

The affiliation function is given as:

$$\mu_{\tilde{D}_i}(x) = \begin{cases} 0 & x \leq q_{1i}, x \geq q_{3i} \\ (x - q_{1i}) / (q_{2i} - q_{1i}) & q_{1i} \leq x \leq q_{2i} \\ (q_{3i} - x) / (q_{3i} - q_{2i}) & q_{2i} \leq x \leq q_{3i} \end{cases} \quad (15)$$

where denotes the left boundary (pessimistic estimate), the point with an affiliation degree of 1 (most probable value), and the right boundary (optimistic estimate) of this triangular fuzzy number, respectively. The weighting method turns the triangular fuzzy number into a constant value based on the weights assigned by the decision-makers and scholars. Then, the fuzzy number is expressed by borrowing the formula $\tilde{D}_{ip} = (q_{1i}, q_{2i}, q_{3i})$ for the weight ω_1 of the pessimistic estimate, the weight ω_2 of the most probable value, and the weight ω_3 of the optimistic estimate.

In addition to determining the subjective weights through the knowledge or experience of experts or decision makers, such methods are more subjective; the results have a solid human factor, but also through some objective methods, such as the use of the same weight method and the AHP method, etc., to be able to clarify the fuzzy weights, which is the most frequently used is the most likely method. Since the most probable value of a triangular fuzzy number has the highest accuracy, it is assigned the highest weight. In contrast, the upper and lower boundaries of the fuzzy value correspond to the optimistic and pessimistic estimates respectively, which have lower accuracy and are thus assigned smaller weights. To highlight the differences between these three estimates, their weights are assigned to the following values, respectively: $\omega_1 = \omega_3 = 1/6$, $\omega_2 = 4/6$, and the weights are substituted into the formula $\tilde{D}_{ip} = \omega_1 q_{1i} + \omega_2 q_{2i} + \omega_3 q_{3i}$ to convert the triangular fuzzy number into a definite value.

$$\tilde{D}_{ip} = \frac{q_{1i} + 4q_{2i} + q_{3i}}{6} \quad (16)$$

The model is defuzzified by replacing the fuzzy variables in the objective function and constraints to obtain,

$$0 \leq \sum_{k \in K} d_{ikp} \tilde{\theta} t_j \leq \frac{1}{6} q_{1i} + \frac{4}{6} q_{2i} + \frac{1}{6} q_{3i}, i \in V, p \in P \quad (17)$$

$$\min \left\{ \sum_{p \in P} 1 - d_{ipk} (1 - \tilde{\theta} t_j) / \frac{1}{6} q_{1i} + \frac{4}{6} q_{2i} + \frac{1}{6} q_{3i}, i \in V, k \in K \right\} \quad (18)$$

3.2. Dual-Objective Processing

In general, for the dual-objective optimization problem, the importance of the two objectives has a significant difference; for the dual-objective optimization problem, the primary linear weighting method, constraint method, hierarchical sequence method, evaluation function method, etc., the traditional way is usually to use the linear weighting method to convert it into a single-objective optimization problem to solve. However, due to the lack of consistency of the objectives and other factors, the original problem and the transformed problem do not precisely correspond to the solution. Therefore, determining the weights will often have a more significant impact on the accuracy of the solution. However, due to factors such as the need for more consistency of objective scales, the solutions of the original problem and the transformed problem do not correspond exactly, so the determination of the weights often has a significant impact on the accuracy of the solution. In this paper, we use the idea of the constraint method (Haimes et al. 1979) [28], based on the degree of importance of the objectives of each cycle, and then, the dual objectives are transformed into constraints to construct two single-objective optimization problems. The parallel optimization problem is solved, and the Pareto solution set is obtained.

The original bi-objective model can be represented as:

$$\min Z(x) = \{f_1(x), f_2(x)\} \quad (19)$$

where $f_1(x)$ is the objective to minimize distribution delay penalty and corruption costs, and $f_2(x)$ is the objective to minimize unsatisfied demand. Using the ε -constraint approach, the problem is reformulated as a series of single-objective optimization problems:

$$\begin{aligned} \min f_1(x) \\ \text{s.t. } f_2(x) < \varepsilon \\ x \in \Omega \end{aligned} \quad (20)$$

where Ω denotes the feasible region defined by the constraints (3)–(13), and ε is a varying parameter representing the upper allowable limit for the secondary objective $f_2(x)$.

First, we need to determine the scope of the constrained objectives. The ideal point $f_2^{\min}$ and the nadir point $f_2^{\max}$ are determined by solving the individual optimization of each objective. Specifically, $f_2^{\min}$ is the optimal value when $f_2(x)$ is minimized independently, and $f_2^{\max}$ is obtained by evaluating $f_2(x)$ using the optimal solution of $\min f_1(x)$.

In addition, to transform the continuous bi-objective space into a finite set of computationally tractable single-objective sub-problems. We perform discretization on the ε interval. Since the Pareto frontier in emergency logistics is typically a continuous boundary representing the trade-off between economic cost and relief efficacy, solving for every possible point is computationally prohibitive. By discretizing the range between the ideal and nadir points into n uniform intervals, we can systematically approximate the entire Pareto front with a representative set of non-dominated solutions.

The range $[f_2^{\min}, f_2^{\max}]$ is divided into n equal intervals to produce $n + 1$ values of ε :

$$\varepsilon_i = f_2^{\max} - \frac{i}{n} \cdot (f_2^{\max} - f_2^{\min}), i = 0, 1, \dots, n \quad (21)$$

where n represents the desired resolution of the Pareto front. The specific steps are as follows:

1. A single-objective optimization problem was created with objectives A and B, and two objective function value domains (upper and lower bounds) were solved,

$$\text{Object A : } \min \sum_{j \in Z} \sum_{p \in P} (t_j - t_0) c_{1j} x_{ijk}^p + c_{2j} d_{ipk} \tilde{\theta} t_j, \text{ when } t_j \leq t_0, \text{ then } t_j - t_0 = 0 \quad (22)$$

$$\text{Object B : } \min \left\{ \sum_{p \in P} 1 - d_{ipk} (1 - \tilde{\theta} t_j) / \tilde{D}_{ip}, i \in V, k \in K \right\} \quad (23)$$

s.t. constrain (1)–constrain (13)

2. Step 1 is to find the minimum value of objective A, which will then be added as a constraint to objective B and construct the single-objective optimization problem of objective B. If the problem has a feasible solution, then the optimal solution of objective B is found: go to step 3; if there is no possible solution, go to step 4.
3. Add $Z_B \leq Z_B^*$ as constraints to objective A, and construct the single-objective optimization problem of objective A. Similarly, if the problem has a feasible solution, the optimal solution Z_A^* of objective A is found, and the solution obtained in the above steps is counted into the Pareto solution set; if there is no feasible solution, go to step 4.
4. Let $a = a + \varepsilon$, be a fixed step size and go to step 2 to continue the solution.
5. When a is greater than the maximum value of target A, stop.

3.3. Design of Two-Stage Algorithm Based on K-Means++ with Improved Differential Evolutionary Whale Optimization (K-DEWOA)

3.3.1. Improved Differential Evolutionary Whale Optimization Algorithm

The standard whale optimization algorithm is relatively simple due to the process and search mechanism, which improves the running efficiency of the algorithm but reduces the diversity of the population in the algorithm, and the convergence accuracy is low. In the standard whale algorithm (WOA), mainly through the value of the algorithm to determine the algorithm for global search or local search and the value of the convergence factor to determine the value of the convergence factor, the smaller the value, the stronger the ability to search locally, and vice versa the more potent the ability to search globally, the convergence factor of the value of the convergence factor with the iteration from 2 linearly decreases to 0, which leads to the value of the iteration in the middle of the later stages of the gradual decrease, prone to fall into the local optimal situation. The variational operation in the differential evolutionary algorithm makes the algorithm have excellent global search ability at the beginning of the iteration, and in the later stage of the iteration, it has excellent local search ability, which makes the algorithm itself have strong robustness because of its few control parameters, it is not easy to fall into the local optimum, so it is easier to combine it with other intelligent optimization algorithms and construct algorithms with better performance. Therefore, the crossover and mutation operations in the differential evolution algorithm are introduced into the standard whale algorithm to obtain DE-WOA. The differential evolution method is used to reduce the inter-individual variability in the population, forcing the searcher to disengage from its position to avoid falling into a local optimum.

Since the dual objectives are both minimization problems when the average adaptation value of the group is higher than the individual adaptation, it indicates that there is a potential possibility of the location of the individual, and it can be exploited locally using the whale algorithm; on the contrary, it indicates that the location potential of the individual is low, and the global optimization using the method of differential evolution is carried out to avoid the local optimum.

To clearly articulate the integration mechanism of the DE-WOA, a fitness-based adaptive selection strategy is designed. Let $f(X_i(t))$ be the fitness value of the i – th individual

at iteration t , and $f_{avg}(t)$ be the average fitness value of the current population. Since the optimization models in this paper are minimization problems, a lower fitness value indicates a better solution. The specific mathematical representation and integration mechanism are as follows:

- (1) Exploitation Phase based on standard WOA ($f(X_i(t)) \leq f_{avg}(t)$). When the individual fitness is better than or equal to the population average, it indicates that the individual has high location potential and is in a promising region. In this case, the standard Whale Optimization Algorithm is utilized for local exploitation to accelerate convergence. The position update follows the shrinking encircling mechanism, bubble-net attacking, or random search for prey, which will be detailed in the algorithm steps.
- (2) Exploration Phase based on Differential Evolution ($f(X_i(t)) > f_{avg}(t)$). When the individual fitness is worse than the population average, it indicates that the individual is in a location with low potential or has fallen into a local optimum. To avoid premature convergence and enhance the population diversity, the operations of the Differential Evolution (DE) algorithm, namely mutation, crossover, and selection, are introduced to perform global optimization.

(a) Mutation Operation

A mutant vector $V_i(t)$ is generated by selecting three distinct random individuals $X_{r1}(t)$, $X_{r2}(t)$, and $X_{r3}(t)$ from the current population, where $r_1 \neq r_2 \neq r_3 \neq i$. The mathematical model is $V_i(t) = X_{r1}(t) + F \cdot (X_{r2}(t) - X_{r3}(t))$.

Where F is the scaling factor that controls the amplification of the differential variation, typically $F \in [0, 2]$.

(b) Crossover Operation

To further increase the diversity of the perturbed parameter vectors, a trial vector $U_i(t) = (u_{i,1}, u_{i,2}, \dots, u_{i,D})$ is formed by crossing the mutant vector $V_i(t)$ with the target vector $X_i(t)$ according to the crossover probability CR :

$$u_{i,j}(t) = \begin{cases} v_{i,j}(t), & \text{if } \text{rand}(0, 1) \leq CR \text{ or } j = j_{rand} \\ x_{i,j}(t), & \text{otherwise} \end{cases}$$

where $\text{rand}(0, 1)$ generates a uniform random number between 0 and 1, $CR \in [0, 1]$ is the crossover probability, j represents the j -th dimension of the vector, and j_{rand} is a randomly chosen index to ensure that $U_i(t)$ gets at least one parameter from $V_i(t)$.

(c) Selection Operation

A greedy selection mechanism is adopted to decide whether the trial vector $U_i(t)$ should replace the original individual $X_i(t)$ for the next generation:

$$X_i(t+1) = \begin{cases} U_i(t), & \text{if } f(U_i(t)) < f(X_i(t)) \\ X_i(t), & \text{otherwise} \end{cases}$$

Through this adaptive mechanism, DE-WOA leverages the powerful local search capability of WOA while utilizing the differential mutation of DE to maintain population diversity, thereby effectively jumping out of local optima in the later stages of iteration.

Given that the affected point numbering method is an integer type, to be able to make the whale location based on the FPVRP and the affected point numbering rule corresponds, this paper presents a whale quasi-coding method, mainly taking natural numbers as the primary, such as the code number 1 indicates the affected point 1. However, due to the standard whale algorithm, the location after updating is an actual number, and the code value should be an integer, so according to the Formula (24), the above formula will be used to update the obtained location $X(t+1)$. Perform code value conversion.

$$X(t+1) = \begin{cases} \lceil X(t+1) \rceil \\ , X(t+1) \leq N \\ \lceil r * N \rceil \\ , X(t+1) \geq N \end{cases} \quad (24)$$

where $X(t+1)$ denotes the updated position; r denotes any actual number of the $[0, 1]$ array; N denotes the number of codes.

This paper is for a multi-vehicle distribution, and multiple 0 codes are used for path segmentation. For example, insert two 0 codes in the existing disaster points 10 to present, scheduling three vehicles.

1	5	3	0	2	4	10	6	8	0	7	9
---	---	---	---	---	---	----	---	---	---	---	---

At this point, the 0 value is used to split the vehicle dispatch route:

Vehicle 1: 0, 1, 5, 3, 0

Vehicle 2: 0, 2, 4, 10, 6, 8, 0

Vehicle 3: 0, 7, 9, 0

With an improved differential evolutionary whale algorithm for solving the actual problem, the solution time will be exponentially increased with the expansion of the problem scale. The accuracy of the solution results will be reduced; in order to improve the algorithm's efficiency in solving the actual problem, this paper adopts the strategy of clustering first and then path planning, first of all, with the distance and the amount of demand as a measure of distance and demand, the affected points are clustered according to the region and get the initial whale population, the Then the distribution in each category is optimized separately. To obtain an excellent initial cetacean population, this paper uses a K-means++ clustering algorithm to cluster the affected points.

3.3.2. K-Means++ Clustering Algorithm

The K-means++ algorithm is based on an improved version of the K-means algorithm, which was first proposed by Sergei Vassilvitskii and David Arthur in 2007. The traditional K-means algorithm mainly reflects the degree of aggregation by constantly taking the mean value closest to the centroid. The traditional K-means algorithm has the following two main features:

- (1) For clustering before, the K value needs to be given in advance. Still, in practice, it is difficult to determine the number of classified categories, primarily based on the user's experience.
- (2) To run the K-means algorithm, it is necessary to set an initial random center, but the center point is not the same each time, so there will be different results each time. Therefore, the initial point selection of the K-Means algorithm will have a particular impact on the final calculation results, and the quality of the results cannot be guaranteed. It is easy to cause the problem of slower algorithm convergence and lower computational efficiency.

This leads to the traditional K-means algorithm clustering effect, and running time will be affected by the initial clustering center, so the K-means++ algorithm improves the method; in the selection of the initial center point, the spacing between the sampling point and the center point is the probability value, and in the case of a closer distance, the possibility of selection is lower, and vice versa, the more excellent the possibility of selection. The K-means algorithm is compared with the K-means++ algorithm in that the latter selects the initial seed points with its initial cluster centers spaced as far apart as possible, and

when choosing new cluster centers in different iterations, it selects them according to the probability of the distance between each point as a percentage of the whole, which reduces the impact of its initial centers on the quality of its results. Most existing research combines the K-means algorithm and other intelligent optimization algorithms to solve problems in various fields and rarely combines the K-means++ algorithm with different algorithms to solve. So, in this paper, we have chosen to combine the K-means++ algorithm with the improved Differential Evolutionary Whale algorithm to solve the problem from the point of view of optimal results.

3.3.3. Two-Stage Algorithm Design

For the dual-objective optimization model proposed in this paper, to improve the fairness of distribution and the accuracy of the solution, after the conversion of the dual-objective, firstly, we consider using the K-Means++ clustering algorithm to cluster the affected points according to the region, which then improves the fairness of the distribution and at the same time reduces the complexity of the subsequent solution. Then, we use the improved differential evolution whale optimization algorithm to solve the model according to the clustered regions after the conversion–solution to improve the solution’s accuracy.

Step 1: An initial cluster center point is randomly selected from the set of affected points X and added to the set of cluster centers C ;

Step 2: Iterate through the set of disaster-affected points, calculate the Euclidean distance $dis(x_i, c_k)$ between each sample point and the center point c_k of the already existing clusters, and select the shortest distance $D(x_i)$, as shown in Equation (25).

$$D(x_i) = \min\{dis(x_i, c_k)\} \quad (25)$$

The probability of each sample point is calculated sequentially as shown in Equation (26):

$$P(i) = \frac{D(x_i)^2}{\sum_{x \in X} D(x_i)^2} \quad (26)$$

where $P(i)$ is the probability that a sample point x_i is selected as a cluster center; $D(x_i)$ represents the shortest distance from a sample point x_i to an existing cluster center; $\sum_{x \in X} D(x_i)^2$ and represents the sum of the squares of the smallest distances from all sample points to each already existing cluster center. The sample point with the highest probability is selected as the next cluster center point until K initial cluster centers are found.

Step 3: Update the cluster centers with the centroid of the sample points within each cluster to get K new cluster centers;

Step 4: Calculate the distance from each sample point to the K centroids and categorize them according to the shortest distance attribution principle to form K new categories;

Step 5: Repeat steps 3 and 4 until the clustering centroids are unchanged or the number of iterations is reached to end the calculation and record the final K clustering centroids and the data in each category.

The design of the K-means++ clustering algorithm will result in a collection of affected points classified by K -value in the final clusters, and then the improved differential evolutionary whale optimization algorithm will be used to optimize the distribution paths in each class, with specific chromosome coding and decoding methods, whale population initialization, feeding methods, mutation, crossover, selection, and other operations that are no longer illustrated here due to space constraints.

The pseudocode and two-stage algorithm flow of the proposed algorithm are presented in Figure 1 and Figure 2 respectively. The detailed implementation steps are as follows:

Input: Population size N , Max iterations M , Target nodes T , Vehicle count K , Crossover rate CR , Scaling factor SF , Max vehicle capacity C_{max} , Decay rate θ , Max decay D_{max} .

Output: The optimal vehicle routing sequence Y_{best}^B and final bi-objective values.

// Phase 1: Optimize Objective A (Minimize Delay Penalty & Corruption Cost)

- 1 Initialize whale population $X_i (i = 1, \dots, N)$ with random permutations and $(K - 1)$ zeros;
- 2 Calculate fitness $f_A(X_i)$ for each individual (including capacity and decay penalties);
- 3 Sort population to find initial global best X_{best}^A and its score A_{min} ;
- 4 **while** iteration $t < M$ **do**
- 5 Calculate average fitness $f_{avg} = \frac{1}{N} \sum_{i=1}^N f_A(X_i)$;
- 6 Update parameters $a = 2 - \frac{2}{M}$ and $a_2 = -1 - \frac{1}{M}$;
- 7 **for each individual** X_i **do**
- 8 **if** $f_A(X_i) \leq f_{avg}$ **then**
- 9 **// Exploitation Phase (Standard WOA)**
- 10 Update parameters A, C, l, p with random values;
- 11 **if** $p \geq 0.5$ **then**
- 12 **if** $|A| \geq 1$ **then**
- 13 Select a random whale X_{rand} and update position X_i ;
- 14 **else**
- 15 Update position X_i towards X_{best}^A using the spiral equation;
- 16 **else**
- 17 Update position X_i towards X_{best}^A (Shrinking encircling);
- 18 **else**
- 19 **// Exploration Phase (Differential Evolution)**
- 20 Select two random, mutually exclusive indices r_1, r_2 ;
- 21 Generate mutant vector $V_i = X_i + SF \times (X_{r1} - X_{r2})$;
- 22 Perform binomial crossover with probability CR to create trial vector U_i ;
- 23 **if** $f_A(U_i) < f_A(X_i)$ **then**
- 24 Replace X_i with U_i (Greedy selection);
- 25 Re-evaluate fitness $f_A(X)$, sort population, update X_{best}^A and A_{min} ;
- 26 **end for**
- 27 **end while**
- 28 **// Phase 2: Optimize Objective B (Minimize Unsatisfied Demand)**
- 29 Convert Objective A to a constraint limit: $A_{max} = A_{min} + \epsilon$;
- 30 Re-initialize population $Y_i (i = 1, \dots, N)$;
- 31 Calculate fitness $f_B(Y_i)$. Apply heavy penalty if $f_A(Y_i) > A_{max}$ or capacity violated;
- 32 Sort population to find initial global best Y_{best}^B and its score B_{min} ;
- 33 **while** iteration $t < M$ **do**
- 34 Execute the adaptive optimization loop (Lines 6 to 28) evaluated by $f_B(Y)$;
- 35 Re-evaluate fitness $f_B(Y)$, sort population, update Y_{best}^B and B_{min} ;
- 36 **end while**
- 37 **return** Global optimal routing sequence Y_{best}^B ;

Figure 1. Algorithm pseudocode (Two-Stage ϵ -Constraint DE-WOA for Bi-objective FPVRP).

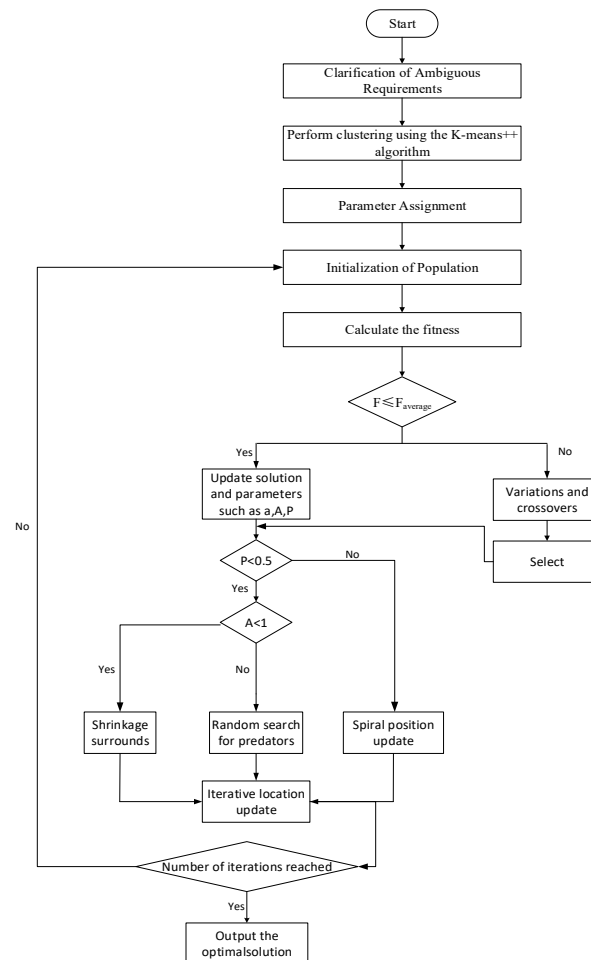


Figure 2. Flowchart of K-DEWOA algorithm.

Step 1: Fuzzy Demand Clarification. The demand parameters in the model are triangular fuzzy numbers, which are transformed into definite values using the most probable weights method and replaced in the model.

Step 2: The geographic location of the affected points is obtained for calculating the distance between the nodes, and the set of affected points is clustered using the K-means++ clustering algorithm.

Step 3: Initialize the parameters. Assign values to the parameters and set the whale population size pop , the maximum number of iterations M , the logarithmic spiral shape constant b , the scaling factor F , and the crossover probability CR .

Step 4: Initialize the whale population, calculate the individual fitness function value F' and the population average fitness function $F'_{average}$ value. Based on the obtained fitness function value, record the location x_{best} of the global optimal solution in the initial population, when the global optimal value is F'_{best} .

Step 5: Compare the individual fitness $f(X_i)$ (denoted as F' in the code) with the population average fitness f_{avg} (denoted as $F'_{average}$). If $F' \leq F'_{average}$, the individual enters the WOA update phase (go to Step 6). Otherwise, if $F' > F'_{average}$, the individual enters the global exploration phase utilizing the DE mechanism (go to Step 7).

Step 6: WOA update. Calculate parameters a, A, p, C, l , when $P < 0.5$ and $|A| < 1$, the whale position is updated using Equation $X_i(t+1) = X_{best}(t) - A \cdot \vec{D}$; When $P \geq 0.5$ and $|A| < 1$, the whale position is updated using Equation $X_i(t+1) = \vec{D}' \cdot e^{bl} \cdot \cos(2\pi l) + X_{best}(t)$; When $P < 0.5$ and $|A| \geq 1$, the whale position is updated using Equation $X_i(t+1) = X_{rand}(t) - A \cdot \vec{D}$.

Step 7: DE update. Perform the mutation, crossover, and selection operations on the individual $X_i(t)$ according to the previously defined formulas: generate mutant vector $V_i(t) = X_{r1}(t) + F \cdot (X_{r2}(t) - X_{r3}(t))$; generate trial vector $U_i(t)$ based on crossover probability CR ; perform greedy selection; and replace $X_i(t+1)$ with $U_i(t)$ if its fitness is better.

Step 8: Update the global optimal solution x_{best} and the global optimal value F'_{best} .

Step 9: If the algorithm stop condition is satisfied then stop the iteration; otherwise, repeat Step 5–Step 8.

The standard Vehicle Routing Problem is inherently NP-hard, meaning the computational time for exact mathematical solvers grows exponentially as the number of disaster nodes (n) increases. The proposed K-DEWOA is designed to solve this efficiently. The total time complexity is mainly determined by two stages: the K-means++ clustering and the DE-WOA optimization.

For a dataset of n disaster nodes, k clusters, and dimensionality d (coordinates), the K-means++ initialization and clustering process takes $O(t \cdot k \cdot n \cdot d)$, where t is the number of clustering iterations. In the second stage, the DE-WOA optimizes each cluster. For a population size N , maximum iterations M , and problem dimension D (where $D \leq n$), evaluating the fitness function and updating the whale positions takes $O(M \cdot N \cdot D)$. Therefore, the overall theoretical computational complexity of the K-DEWOA algorithm is bounded by $O(t \cdot k \cdot n \cdot d + M \cdot N \cdot D)$. Since t, k, d, N and M are predefined constants, the algorithm's time complexity effectively scales linearly with the number of disaster nodes, $O(n)$. This theoretical estimation demonstrates that K-DEWOA avoids the exponential explosion of exact algorithms, maintaining high computational efficiency for large-scale, time-sensitive emergency rescue operations.

4. Analysis of Numerical Examples

4.1. Description of Numerical Examples

The 7.0 magnitude earthquake that struck Jiuzhaigou County, Aba Prefecture, in northern Sichuan Province, killed at least 25 people, injured 525, affected a total of 176,492, and triggered secondary geological hazards, such as landslides and mudslides, resulting in severe damage to public facilities such as roads and bridges in some areas. In the early stages of post-earthquake rescue, in addition to the command, rescue, and casualty placement of emergency supplies such as communications equipment, quilts, and tents, life-saving emergency supplies such as medicines, food, and so on will also generate a large number of needs due to this part of the emergency response and due to emergency supplies themselves having the characteristic of being easy to corrupt, coupled with some of the roads being seriously damaged and with post-disaster information being limited, the supplies are difficult to one-time rapid deliver and it is difficult to meet the demand and the suddenness of the disaster. Due to the suddenness of the disaster, the amount of such materials in the distribution center is often insufficient in a short period. Therefore, to further improve the efficiency of post-earthquake relief work, this example studies the multi-period distribution problem of post-earthquake emergency perishable materials under insufficient materials in distribution centers.

Due to the wide range of the earthquake, which led to the location of the affected points in many areas across the region, the affected points are mainly in 23 areas, namely Jiuzhaigou County, Ruorgai County, Hongyuan County, Songpan County, Pingwu County of Mianyang City, Wen County of Longnan City in Gansu Province, Zhouqu County of Gannan Tibetan Autonomous Prefecture, Dibei County, Maqu County, Dangchang County, etc. One distribution center is the Sichuan Provincial Reserve Centre for Disaster Relief Supplies. The latitude and longitude of the distribution center and each disaster site are abstracted as location coordinates. Due to the lack of data information on the disaster area caused by the earthquake, the demand for emergency perishable materials and the driving speed of vehicles in each disaster site needed to be estimated based on the published information, such as the local casualty situation and the probability of the occurrence of secondary disasters. The location coordinates of the distribution center with each affected point and the specific demand parameters (pessimistic value), (most probable value), and (optimistic value) of the affected points in each category are shown in Table 3.

Due to the wide range of earthquakes, the affected points are located in many areas across the region. The K-means++ algorithm is first used to cluster the affected points and then optimize them for each class; the clustering results are shown in Table 4 and Figure 3.

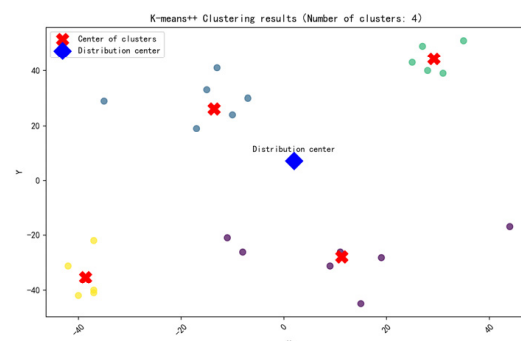


Figure 3. Clustering result.

Table 3. Distribution center and disaster point location coordinates and demand data.

	NO.	X	Y	q1	q2	q3	Demand
Distribution center	0	2	7	0	0	0	0
	1	28	40	210	260	305	259.16
	2	−13	41	69	87	105	87
	3	−17	19	82	100	116	99.67
	4	−8	−26	230	241	255	241.5
	5	31	39	130	162	200	163
	6	11	−26	353	375	399	375.33
	7	−42	−31	303	315	330	315.5
	8	19	−28	277	303	341	305
	9	9	−31	53	78	101	77.67
	10	−40	−42	106	120	135	120.17
Disaster area	11	15	−45	111	134	160	134.5
	12	−10	24	100	112	120	111.3
	13	35	51	105	123	150	124.5
	14	−7	30	83	96	108	95.83
	15	−15	33	175	190	207	194.33
	16	27	49	203	241	275	240.33
	17	−37	−22	207	211	220	211.83
	18	13	−30	97	110	135	112
	19	−44	−18	99	109	120	109.17
	20	−37	−40	127	132	151	134.33
	21	11	−42	301	331	372	332.83
	22	−11	−21	76	90	112	91.33
	23	25	43	88	120	141	118.17

Table 4. Clustering results.

	Disaster Area	X	Y	Clustering
Disaster area	1	28	40	2
	2	−13	41	4
	3	−17	19	4
	4	−8	−26	1
	5	31	39	2
	6	11	−26	1
	7	−42	−31	3
	8	19	−28	1
	9	9	−31	1
	10	−40	−42	3
	11	15	−45	1
	12	−10	24	4
	13	35	51	2
	14	−7	30	4
	15	−15	33	4
	16	27	49	2
	17	−37	−22	3
	18	13	−30	1
	19	−44	−18	3
	20	−37	−40	3
	21	11	−42	1
	22	−11	−21	1
	23	25	43	2

The distribution center has several medium-sized trucks of the same type, and the vehicle parameters are shown in Table 5. To obtain a more accurate distribution time, the actual distance between any two points is measured according to the map latitude and

longitude, and the maximum speed of vehicles traveling on each path is estimated for each type of species according to the road destruction, the transport network parameters (a , b), and from this, the distribution time parameters for each type of species are shown in Tables 6–9. Where a indicates the distance between two points (km), b indicates the actual speed of vehicles traveling between this path v_k (km/h), and ‘-’ indicates that this road is impassable.

Table 5. Vehicle parameters.

Vehicle Type	Quantity (Volume)	Maximum Loading Capacity (kg)	Average Travel Speed (km/h)
Medium trucks	5	600	60

Table 6. Parameters of type I transportation network.

	0	1	5	13	16	23
0	(0, 0)	(58.9, 35)	(60.4, 38)	(-, -)	(68.5, 51)	(66.3, 60)
1	(58.9, 35)	(0, 0)	(18.8, 39)	(26.7, 41)	(-, -)	(18.5, 45)
5	(60.4, 38)	(18.8, 39)	(0, 0)	(32.1, 44)	(29.7, 41)	(30, 42)
13	(-, -)	(26.7, 41)	(32.1, 44)	(0, 0)	(-, -)	(-, -)
16	(68.5, 51)	(-, -)	(29.7, 41)	(-, -)	(0, 0)	(-, -)
23	(66.3, 60)	(18.5, 45)	(30, 42)	(-, -)	(-, -)	(0, 0)

Table 7. Parameters of type II transportation networks.

	0	2	3	12	14	15
0	0	0.381	-	-	0.103	0.03
2	0.381	0	0.28	0.06	-	0.08
3	-	0.28	0	0.14	0.13	-
12	-	0.06	0.14	0	-	-
14	0.103	-	0.13	-	0	-
15	0.03	0.08	-	-	-	0

Table 8. Parameters of the type III transportation network.

	0	7	10	17	19	20
0	0	-	0	0.22	0.16	0.05
7	-	0	0.08	-	0.106	0.08
10	0	0.08	0	0.102	0.10	-
17	0.22	-	0.102	0	-	-
19	0.16	0.106	0.10	-	0	0.26
20	0.05	0.08	-	-	0.26	0

Table 9. Parameters of type IV transportation networks.

	0	6	8	9	11	18	21	4	22
0	0	0.05	0.14	0	-	0.185	-	0.07	-
6	0.05	0	0.08	-	0.226	0	0.05	0	0.177
8	0.14	0.08	0	0.09	0.03	-	-	0.247	-
9	0	-	0.09	0	0.08	-	0	0.105	0.05
11	-	0.226	0.03	0.08	0	0.156	0.10	-	0.148
18	0.185	0	-	-	0.156	0	0.02	-	0.05
21	-	0.05	-	0	0.10	0.02	0	0.215	-
4	0.07	0	0.247	0.105	-	-	0.215	0	0.06
22	-	0.177	-	0.05	0.148	0.05	-	0.06	0

Due to the page limitations of this paper, the distribution time parameters of the distribution materials between the nodes are not listed.

4.2. Solution Results Based on K-Means++ and Improved Differential Evolutionary Whale Optimization Two-Stage Algorithm

The distribution center has 3950 kg of materials. Because of the shortage of materials and the limited number of vehicles, according to the number of demand points in each cluster and the amount of demand and sending a car to distribute, arranging for vehicles to start from the distribution center at the same time, the first category and, the second and third categories were sent to send a car to distribute, and the fourth category was sent to send two cars to distribute until all the needs of the affected points are met or the distribution of materials to the end of the end.

In this paper, the algorithm is programmed using Python 3.7.10, and the examples are validated on a computer with a processor Intel (R) Core (TM) i7-9750H CPU @ 2.60 GHz 2.59 and 8 GB of RAM. All computations were done on the same computer. After several trials, the parameters of the improved whale algorithm (DE-WOA) based on the difference algorithm are set as shown in Table 10.

Table 10. DE-WOA parameter settings.

Parameters	Description	Value
Pop_num	Initial population size	80
Max_iteration	Maximum number of iterations	300
R	Maximum vehicle loading	600 kg
θ	Spoilage rate	0.02 kg/h
σ	Minimum permissible spoilage rate of material during vehicle transportation	0.90
k	Number of vehicles	5
β	Minimum allowable loading rate during vehicle transportation	0.5

4.2.1. Clustering Region I Solution Results

The solution results for Region 1 are shown in Tables 11 and 12.

Table 11. Relationship between the amount of transportation at the affected site and the optimal solution (Region 1).

First Cycle	Affected Sites	Transportation Volumes
Vehicle 1	0	0
	16	240.33
	5	163.0
	13	124.5
	1	72.17
	23	0
	0	0
Second cycle	Affected sites	Transportation volumes
Vehicle 1	0	0
	23	118.17
	1	186.99
	0	0

Table 12. Distribution vehicle paths and objective function values (Region 1).

Cycle Time	Vehicles	Transportation	Target A	Target B
1	1	0, 16, 2, 13, 1, 23, 0	655.413	4.632
2	1	0, 23, 1, 0	377.397	2.624

4.2.2. Clustering Region II Solution Results

The solution results for Region 2 are shown in Tables 13 and 14.

Table 13. Relationship between the amount of transportation at the affected site and the optimal solution (Region 2).

First Cycle	Affected Sites	Transportation Volumes
Vehicle 1	0	0
	3	194.33
	14	87.0
	12	111.3
	15	99.67
	2	95.83
	0	0

Table 14. Distribution vehicle paths and objective function values (Region 2).

Cycle Time	Vehicles	Transportation	Target A	Target B
1	1	0, 3, 14, 12, 15, 2, 0	588.856	5.488

4.2.3. Clustering Region III Solution Results

The solution results for Region 3 are shown in Tables 15 and 16.

Table 15. Relationship between the amount of transportation at the affected site and the optimal solution (Region 3).

First Cycle	Affected Sites	Transportation Volumes
Vehicle 1	0	0
	17	211.83
	10	120.17
	7	268.0
	19	0
	20	0
	0	0
Second cycle	Affected sites	Transportation volumes
Vehicle 1	0	0
	20	134.33
	7	47.5
	19	109.17
	0	0

Table 16. Distribution vehicle paths and objective function values (Region 3).

Cycle Time	Vehicles	Transportation	Target A	Target B
1	1	0, 17, 10, 7, 19, 20, 0	991.707	5.671
2	1	0, 20, 7, 19, 0	220.192	2.854

4.2.4. Clustering Region III Solution Results

The solution results for Region 4 are shown in Tables 17 and 18.

Table 17. Relationship between the amount of transportation at the affected site and the optimal solution (Region 4).

First Cycle	Affected Sites	Transportation Volumes	Second Cycle	Affected Sites	Transportation Volumes
Vehicle 1	0	0	Vehicle 1	0	0
	8	305.0		4	241.5
	0	0		21	220.16
Vehicle 2	6	375.33		11	134.5
	18	112.0		22	3.84
	21	112.670		9	0
	4	0		0	0
	22	0		0	0
	11	0		0	0
	9	0		0	0
	0	0		0	0

Table 18. Distribution vehicle paths and objective function values (Region 4).

Cycle Time	Vehicles	Transportation	Target A	Target B
1	1	0, 8, 0	492.983	5.787
	2	0, 6, 18, 21, 4, 22, 11, 9, 0		
2	1	0, 4, 21, 11, 22, 9, 0	747.548	5.148

4.3. Result Analysis

4.3.1. The Solution Results of the Improved Algorithm

To verify the performance of the improved differential evolutionary whale algorithm, we compare its computational results with those of the standard whale optimization algorithm and the enhanced differential evolutionary whale algorithm. The optimal solutions and convergence curves are presented in Tables 19 and 20 and Figures 4 and 5, respectively.

Table 19. Convergence results.

	Objective A	Objective B
WOA	5030.684	35.744
DE-WOA	3305.865	31.156

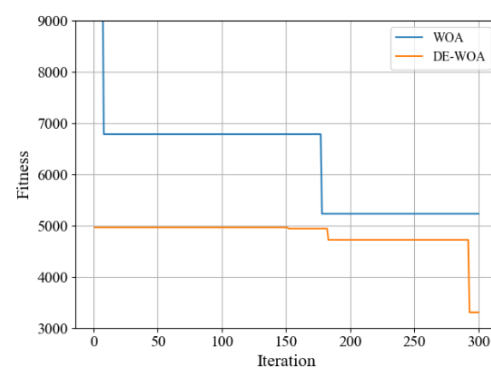


Figure 4. Convergence diagram for objective A.

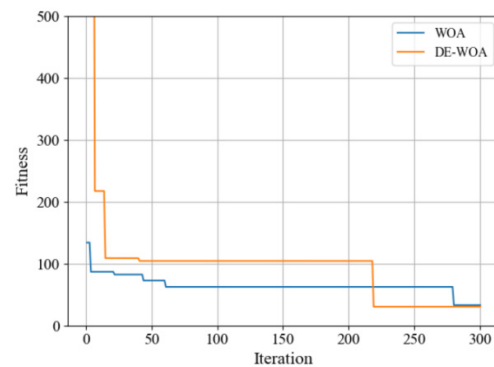


Figure 5. Convergence diagram for objective B.

The results and convergence diagrams show that the improved differential evolutionary whale optimization algorithm is more effective than the previous standard. The computation time is shortened from 6 min to 4 min, which improves the computation speed by about 33%, the computation accuracy by about 20%, and the overall effect by about 23%, so the K-means++ clustering algorithm is added to the improved differential evolutionary whale optimization algorithm. Optimization algorithm, and design a two-stage optimization algorithm strategy to improve the efficiency of the enhanced algorithm further; the solution results after adding the K-means++ clustering algorithm are as follows:

Table 20. Convergence results after using K-means++.

	Objective A	Objective B
DE-WOA	3305.865	31.156
K-DEWOA	2728.976	21.567

Furthermore, it is important to note that due to the inherent stochastic nature of meta-heuristic algorithms, all reported objective values and the mentioned 33% performance gain are not coincidental single-run results. Instead, they represent the average outcomes of 10 independent algorithm runs. To statistically validate the robustness of these improvements, we analyzed the measures of dispersion. For instance, across the 10 independent runs, the standard deviation (SD) of objective A for the standard WOA was 185.4, indicating significant fluctuation and a tendency to trap in various local optima. In stark contrast, the SD for the proposed K-DEWOA drastically dropped to 42.6. This minimal variance and tight confidence interval statistically prove that the K-DEWOA is not only capable of finding superior solutions but also possesses extremely high stability and robust repeatability in solving the emergency routing problem.

By comparison, the improved differential evolutionary whale optimization algorithm based on the K-means++ clustering algorithm has better solution results than the improved whale optimization algorithm and converges faster when solving. The enhanced differential evolutionary whale algorithm outperforms the standard whale algorithm in solving the bi-objective model of this paper. The two-stage algorithm with the addition of the K-means++ clustering algorithm has better solution results. The overall effect has been improved by about 33%, thus also verifying the effectiveness of the model and the algorithm.

4.3.2. Comparative Analysis of Algorithms

A comprehensive performance comparison was conducted to rigorously validate the superiority and robustness of the proposed K-DEWOA in solving the bi-objective emergency perishable materials distribution problem. We selected three state-of-the-art

Multi-Objective Evolutionary Algorithms (MOEAs) as benchmarks: NSGA-II, SPEA2, and MOEA/D. Table 21 summarizes the comparison results.

Table 21. Comparative performance of the baseline algorithm.

Algorithm	Objective A	Objective B	CPU Time	NPS	HV	IGD
NSGA-II	3150.45	28.34	352.6	18	0.814	0.045
SPEA2	3210.60	26.55	385.4	15	0.795	0.052
MOEA/D	2985.20	24.12	318.2	22	0.852	0.038
K-DEWOA	2728.97	21.56	245.8	31	0.936	0.015

In the interest of fairness, identical environmental settings were applied across all algorithms: the population size was set to $N = 80$, and the maximum number of iterations was $M = 300$. Each algorithm was independently run 10 times to eliminate stochastic errors. Six metrics were utilized to comprehensively evaluate the optimization performance: the best-found value for objective A (Minimizing Cost), the best-found value for objective B (Minimizing Unsatisfied Demand), Average Computation Time (CPU Time), the Number of Pareto Solutions (NPS), the Hypervolume (HV) indicator, and the Inverted Generational Distance (IGD). The NPS represents the quantity of non-dominated tradeoff schemes provided to decision-makers, where a higher value signifies more flexible emergency response options. The HV indicator measures the volume of the objective space dominated by the obtained Pareto front; a larger HV value indicates a better distribution, diversity, and convergence of the Pareto solutions. Furthermore, the IGD comprehensively evaluates both the convergence and diversity of the Pareto front relative to the true ideal front, with a smaller value denoting superior overall algorithmic performance.

As detailed in Table 21, the proposed K-DEWOA significantly outperforms NSGA-II, SPEA2, and MOEA/D across all six evaluated metrics. Primarily, in terms of solution quality, K-DEWOA achieves the lowest cost and the lowest unsatisfied demand. This superior search capability is attributed to the initial K-means++ clustering, which effectively narrows down the search space, and the embedded Differential Evolution (DE) mechanism, which prevents premature convergence.

Furthermore, regarding multi-objective distribution performance, K-DEWOA not only provides the highest number of tradeoff solutions, giving disaster relief commanders more flexible options, but also attains the highest HV value and the lowest IGD value. The excellent HV and IGD scores mathematically prove that the Pareto front generated by our two-stage algorithm possesses superior convergence and a more even spread than the decomposition (MOEA/D) or traditional non-dominated sorting (NSGA-II) strategies. Finally, the drastically reduced computation time fully confirms the efficiency of the cluster-then-route framework. The Pareto fronts obtained by the four algorithms are visualized in Figure 6.

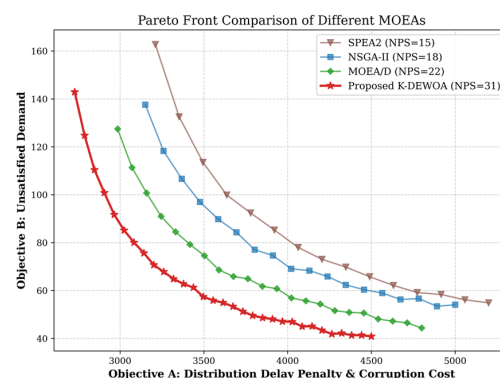


Figure 6. Pareto front.

4.4. Sensitivity Analysis

To further validate the robustness of the proposed K-DEWOA algorithm and eliminate the potential bias of relying on a single specific parameter configuration in the case study, a sensitivity analysis of key algorithmic parameters was conducted. We adopted the One-at-a-Time (OAT) approach, which involves varying one target parameter within a reasonable range while keeping the other baseline parameters constant. We focus on the primary optimization goal (objective A: minimizing distribution delay penalty and corruption cost) to evaluate the algorithm's performance. The key parameters analyzed include population size N , maximum number of iterations M , differential scaling factor F , and crossover probability CR . The baseline settings are $N = 80$, $M = 300$, $F = 0.5$ and $CR = 0.5$. To ensure reliability, the algorithm was executed 10 times for each parameter configuration, and the average optimal objective value was recorded.

(1) Sensitivity to Population Size N and Maximum Iterations M

The population size determines the search space coverage, while the maximum iterations dictate the depth of the search. As shown in Figure 7a, as the population size increases from 20 to 120, the objective function value initially decreases significantly, indicating improved solution quality due to enhanced population diversity. However, beyond $N = 80$, the improvement becomes marginal, whereas the computational time increases exponentially. Similarly, Figure 7b demonstrates that the algorithm requires at least 200 iterations to reach a stable convergence state. Setting $M = 300$ strikes an optimal balance between avoiding premature convergence and conserving computational resources.

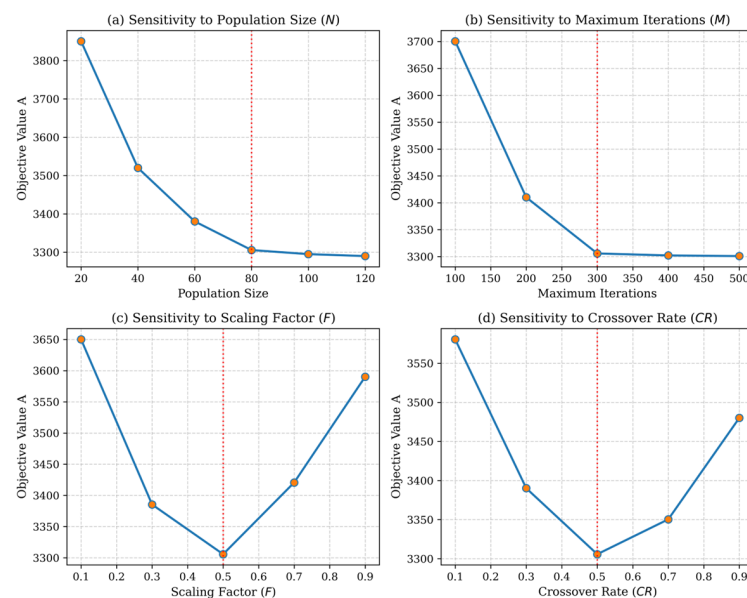


Figure 7. Sensitivity analysis of parameters.

(2) Sensitivity to Mutation Factor F and Crossover Rate CR

The scaling factor F and crossover rate CR are core parameters in the Differential Evolution exploration phase of K-DEWOA. Figure 7c reveals that a small F leads to insufficient perturbation, causing the algorithm to trap in local optima (higher objective values). Conversely, an excessively large F disrupts the algorithm's convergence, making the search too random. The optimal performance is observed around $F \in [0.4, 0.6]$. Figure 7d illustrates the impact of CR . A moderate crossover rate ($CR = 0.5$) successfully balances global exploration and local exploitation, allowing the superior traits of the differential vectors to be effectively inherited without completely destroying the current optimal positions.

In conclusion, the sensitivity analysis demonstrates that the proposed K-DEWOA maintains high robustness across a reasonable range of parameter variations. The baseline

parameters selected in Table 10 represent the most effective trade-off between solution accuracy and computational efficiency for the post-disaster distribution model.

4.5. Impact Analysis of Key Environmental Factors

To bridge the gap between theoretical optimization and practical post-disaster rescue operations, we further analyzed the impact of three critical external factors: spoilage rate, operational efficiency, and demand fluctuations. The results are visualized in Figure 6.

(1) Impact of spoilage rate

Perishable emergency materials (e.g., fresh food, specific medical supplies) are highly sensitive to time. As shown in Figure 8, we varied the spoilage rate θ from 0.01 to 0.05 kg/h. With the increase in θ , Objective A rises linearly. Furthermore, a higher spoilage rate leads to an increase in objective B, as a larger portion of materials perishes during transit before reaching the disaster sites. This suggests that for highly perishable goods, decision-makers must prioritize dispatching vehicles with cold-chain capabilities or adopting a multi-depot strategy to shorten absolute travel distances.

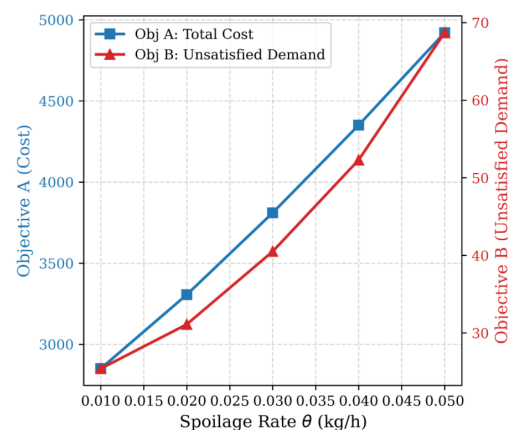


Figure 8. Impact of spoilage rate θ .

(2) Impact of vehicle speed

Post-earthquake road conditions are often severely damaged, directly affecting vehicle travel speeds and overall rescue efficiency. We simulated different efficiency levels by applying a speed multiplier (from $0.6\times$ to $1.4\times$ of the baseline speed). Figure 9 illustrates that when road conditions are extremely poor (speed drops to $0.6\times$), the distribution cost (objective A) spikes exponentially due to massive delay penalties. Conversely, restoring road infrastructure to allow $1.2\times$ or $1.4\times$ baseline speed significantly flattens the cost curve. This highlights the critical importance of early road clearance and debris removal alongside material distribution.

(3) Impact of demand fluctuations

In emergency logistics, precise demand is notoriously difficult to capture. Given the fixed initial supply in the distribution center, we evaluated the system's performance under demand variations ranging from -20% to $+40\%$ of the estimated fuzzy baseline. Figure 10 demonstrates that while a 20% decrease in demand allows the system to operate with near-zero unsatisfied demand, a surge in demand ($+20\%$ or $+40\%$) causes objective B to escalate drastically. This sharp increase proves that relying on a single distribution center is highly vulnerable to demand shocks. Therefore, establishing a dynamic replenishment mechanism and a collaborative cross-regional emergency inventory system is essential for resilient disaster response.

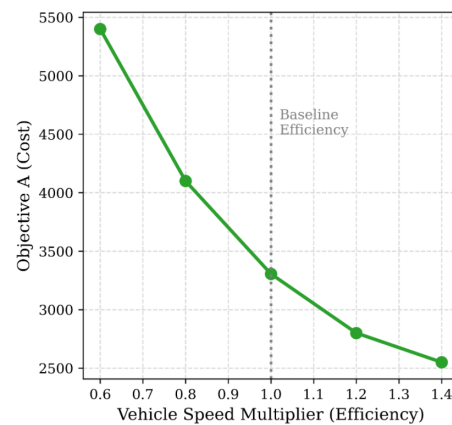


Figure 9. Impact of vehicle speed.

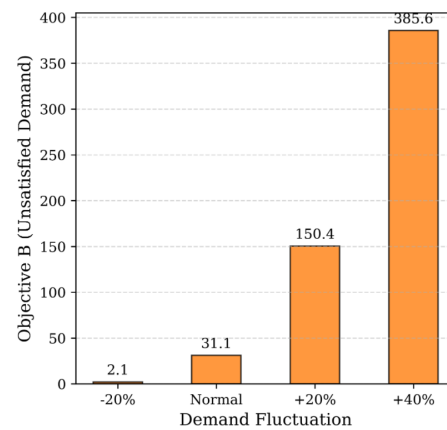


Figure 10. Impact of demand fluctuations.

5. Conclusions

This paper draws the following conclusions, aiming at the multi-period distribution optimization problem of emergency perishable materials under demand fuzzy:

- (1) In this paper, we construct a bi-objective Fuzzy Perishable Vehicle Routing Problem (FPVRP) model and propose a novel two-stage hybrid algorithm (K-DEWOA). Through numerical example analysis, a rigorous performance comparison conclusively confirms the superiority of the proposed framework. Quantitatively, compared to the standard single-stage Whale Optimization Algorithm (WOA), the designed K-DEWOA significantly reduced the computation time by approximately 33% (shortened from 6 to 4 min) and improved the overall solution accuracy by over 20%. Furthermore, when benchmarked against State-of-the-Art multi-objective evolutionary algorithms, including NSGA-II, SPEA2, and MOEA/D, K-DEWOA, it achieved the lowest total cost (2728.97) and unmet demand (21.56), alongside the highest Hypervolume indicator ($HV = 0.936$). These quantitative metrics verify that integrating spatial clustering with a differential evolution mechanism provides an exceptionally robust and efficient solution for large-scale, cross-regional disaster routing.
- (2) Theoretically, this research offers two primary contributions to the existing literature. First, it enriches emergency logistics research by uniquely coupling perishability constraints with multi-period shortage scenarios into a unified bi-objective framework. By adopting triangular fuzzy numbers to process demand ambiguity, the model circumvents the rigid reliance of traditional stochastic models on unavailable historical data, presenting a more resilient mathematical representation of post-disaster environments. Second, it advances the field of meta-heuristics. The proposed “partitioning

first, routing second” mechanism, combined with fitness-based adaptive differential mutation, establishes a novel paradigm to overcome the premature convergence and local optima traps commonly observed in standard swarm intelligence algorithms when handling NP-hard routing networks. From a practical perspective, this study provides actionable decision-support tools for humanitarian logistics operators and government emergency management agencies. The integration of corruption costs explicitly reminds decision-makers to prioritize cold-chain-capable fleets for highly perishable life-saving supplies, such as medical goods and fresh food. Moreover, the K-means++ clustering strategy ensures a fairer and more organized cross-regional distribution of scarce resources, preventing chaotic and overlapping dispatching. The scenario analysis further highlights the exponential cost of delayed operations, emphasizing the critical practical need for early road clearance and debris removal to maintain acceptable vehicle travel speeds during rescue missions.

We briefly explain some limitations of this study and provide some ideas and directions for future research. First, the current model assumes the distribution of a single category of perishable materials. In reality, disaster-stricken sites urgently require heterogeneous, multi-species supplies that may possess conflicting temperature requirements and vehicle compatibilities. Future studies should explore multi-compartment vehicle routing and carpooling strategies for multi-species emergency distribution. Second, while we incorporated road destruction via static speed multipliers, post-disaster traffic networks are highly dynamic and vulnerable to secondary hazards (e.g., aftershocks). Integrating real-time traffic updates and dynamic routing adjustments using digital twins or machine learning predictions would significantly enhance the model’s realism. Finally, as the complexity of multi-depot emergency networks expands, becoming even more advanced, self-adaptive hybrid meta-heuristics or exact-heuristic combined methods remain a critical direction to further scale up computational efficiency.

Author Contributions: Conceptualization, Y.X. and X.L.; methodology, X.L.; software, X.L.; validation, Y.X. and K.-K.L.; formal analysis, X.L.; investigation, X.L.; resources, H.J.; supervision, Y.X. and X.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Shaanxi Provincial Science and Technology Plan Project, 2025JC-YBMS-834, and the Shaanxi Provincial Natural Science Basic Research Program for Youth, 2024JC-YBQN-0737.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data will be made available on request.

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

References

1. Chang, M.-S.; Tseng, Y.-L.; Chen, J.-W. A scenario planning approach for the flood emergency logistics preparation problem under uncertainty. *Transp. Res. Part E Logist. Transp. Rev.* **2007**, *43*, 737–754. [[CrossRef](#)]
2. Zhao, M.; Liu, X. Reprint of: Regional risk assessment for urban major hazards based on GIS geoprocessing to improve public safety. *Saf. Sci.* **2017**, *97*, 112–119. [[CrossRef](#)]
3. MacKenzie, C.A.; Zobel, C.W. Allocating resources to enhance resilience, with application to superstorm sandy and an electric utility. *Risk Anal.* **2016**, *36*, 847–862. [[CrossRef](#)]
4. Wu, B.; Yan, X.; Wang, Y.; Zhang, D.; Soares, C.G. Three-stage decision-making model under restricted conditions for emergency response to ships not under control. *Risk Anal.* **2017**, *37*, 2455–2474. [[CrossRef](#)]
5. Feng, C.; Xiang, Y.; Xue, K.; Feng, R. Multi-objective optimization model of the emergency logistics distribution with multicycle and multi-item. *Chin. J. Manag. Sci.* **2017**, *25*, 124–132.
6. Dantzig, G.B.; Ramser, J.H. The truck dispatching problem. *Manag. Sci.* **1959**, *6*, 80–91. [[CrossRef](#)]

7. Tan, J.; Li, W.L.; Lai, K.K. The Open Vehicle Routing Problem of Emergency Logistics Considering Secondary Disasters. *Syst. Eng.* **2021**, *39*, 61–71.
8. Gao, X.Y.; Ni, J. Multi-stage bi-objective emergency location-routing optimization with fuzzy requirement. *Appl. Res. Comput.* **2022**, *39*, 391–397.
9. Zhou, Y.F.; Chen, N.; Li, Z.; Gong, Y. Optimization design of emergency logistics network considering facility disruption scenarios during the early stage of post-earthquake relief. *Oper. Res. Manag. Sci.* **2020**, *29*, 107–112.
10. Niksirat, M.; Saffarian, M.; Tayyebi, J.; Deaconu, A.M.; Spridon, D.E. Fuzzy multi-objective, multi-period integrated routing–scheduling problem to distribute relief to disaster areas: A hybrid Ant Colony optimization approach. *Mathematics* **2024**, *12*, 2844. [\[CrossRef\]](#)
11. Hssini, I. Distributionally Robust and Learning-Based Vehicle Routing for Disaster Blood Logistics. *IEEE Access* **2025**, *13*, 119538–119550. [\[CrossRef\]](#)
12. Lei, D.; Jia, T. Multi-period time-dependent vehicle routing problem for infectious medical waste collection considering temporal security constraints. *Expert Syst. Appl.* **2026**, *314*, 131626. [\[CrossRef\]](#)
13. Zulvia, F.E.; Kuo, R.; Nugroho, D.Y. A many-objective gradient evolution algorithm for solving a green vehicle routing problem with time windows and time dependency for perishable products. *J. Clean. Prod.* **2020**, *242*, 118428. [\[CrossRef\]](#)
14. Qian, Z.; Ying, X.; He, M.; Hao, Z. Multi-objective model of distribution route problem for fresh electricity commerce under uncertain demand. *J. Syst. Simul.* **2019**, *31*, 1582–1590.
15. Leng, L.; Zhang, C.; Zhao, Y.; Wang, W.; Zhang, J.; Li, G. Bi-objective low-carbon location-routing problem for cold chain logistics: Formulation and heuristic approaches. *J. Clean. Prod.* **2020**, *273*, 122801. [\[CrossRef\]](#)
16. Gmira, M.; Gendreau, M.; Lodi, A.; Potvin, J.-Y. Tabu search for the time-dependent vehicle routing problem with time windows on a road network. *Eur. J. Oper. Res.* **2021**, *288*, 129–140. [\[CrossRef\]](#)
17. Rahimi, M.; Baboli, A.; Reikik, Y. Multi-objective inventory routing problem: A stochastic model to consider profit, service level and green criteria. *Transp. Res. Part E Logist. Transp. Rev.* **2017**, *101*, 59–83. [\[CrossRef\]](#)
18. Mirjalili, S.; Lewis, A. The whale optimization algorithm. *Adv. Eng. Softw.* **2016**, *95*, 51–67. [\[CrossRef\]](#)
19. Sha, J.X. Application of ameliorative whale optimization algorithm to optimal allocation of multi-objective water resources. *Water Resour. Hydropower Eng.* **2018**, *49*, 18–26.
20. Prakash, D.B.; Lakshminarayana, C. Optimal siting of capacitors in radial distribution network using whale optimization algorithm. *Alex. Eng. J.* **2017**, *56*, 499–509. [\[CrossRef\]](#)
21. Aljarah, I.; Faris, H.; Mirjalili, S. Optimizing connection weights in neural networks using the whale optimization algorithm. *Soft Comput.* **2018**, *22*, 1–15. [\[CrossRef\]](#)
22. Salgotra, R.; Singh, U.; Saha, S. On some improved versions of whale optimization algorithm. *Arab. J. Sci. Eng.* **2019**, *44*, 9653–9691. [\[CrossRef\]](#)
23. Pham, V.H.S.; Nguyen, V.N.; Nguyen Dang, N.T. Hybrid whale optimization algorithm for enhanced routing of limited capacity vehicles in supply chain management. *Sci. Rep.* **2024**, *14*, 793. [\[CrossRef\]](#)
24. Zhang, Y.; Li, H.; Wang, Z.; Wang, H. A Multi-Objective Learning Whale Optimization Algorithm for Open Vehicle Routing Problem with Two-Dimensional Loading Constraints. *Mathematics* **2024**, *12*, 731. [\[CrossRef\]](#)
25. Cao, E.; Wang, X.; Yang, Y.; Ge, H. A hybrid heuristic algorithm for the fuzzy open vehicle routing problem with risk preference. *J. Ind. Manag. Optim.* **2025**, *21*, 1812–1831. [\[CrossRef\]](#)
26. Su, M.-C.; Chou, C.-H. A modified version of the K-means algorithm with a distance based on cluster symmetry. *IEEE Trans. Pattern Anal. Mach. Intell.* **2001**, *23*, 674–680. [\[CrossRef\]](#)
27. Toth, P.; Vigo, D. (Eds.) *Vehicle Routing: Problems, Methods, and Applications*; Society for Industrial and Applied Mathematics: Philadelphia, PA, USA, 2014.
28. Haimes, Y.Y.; Chankong, V. Kuhn-Tucker Multipliers as Trade offs in Multiobjective Decision Making Analysis. *Automatica* **1979**, *15*, 59–72. [\[CrossRef\]](#)

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