Current Science
CS 4 (6), 938– 954 (2024)
Received 14 Oct 2024 | **Revised 20 Nov.** 2024 | **Accepted** 05 **Dec**. 2024 | **Online Available 30 Dec** 2024
https://doi.org/10.52845/CS/2024-4-6-17

Current Science
Current Science

OPEN ACCESS

ISSN(O)2795-8639

ORIGINAL ARTICLE

Optimization of Two-echelon Emergency Material Distribution Path Based on Prospect Theory

Houxue Xia[1], Bowen Wu[1], Huagang Tong[1,2], Song Wang[3*]

[1]School of Economics and Management, Nanjing Tech University, Nanjing 210000, China；

[2]School of Economics and Management, Tianjin University, Tianjin 30072, China;

[3]Department of World Economics and Politics, Party School of C.P.C. Jiangsu Provincial Committee, Nanjing 210009, China

*Corresponding Author: Song Wang

Abstract

In emergency situations, addressing the critical issue of diminishing satisfaction stemming from delayed deliveries has emerged as a paramount challenge. To tackle this, we developed a sophisticated hybrid mathematical model that integrates prospect theory with fuzzy mathematics methodologies. Furthermore, we implemented an innovative two-echelon transportation approach to optimize overall transportation efficiency. Lastly, we devised an efficient Hybrid Genetic Algorithm by merging the Genetic Algorithm with an adaptive large neighborhood search algorithm. Comparative experiments and case studies have demonstrated the superior effectiveness of this hybrid algorithm in resolving the issue.

Key words prospect theory; two-echelon vehicles routing problem; hybrid genetic algorithm; emergency situations

Introduction

According to data released by the Global Disaster Data Platform, a staggering 620 disasters struck in the first half of 2023, predominantly natural calamities such as wildfires, floods, and earthquakes. These disasters impacted an astonishing 63,063,200 individuals and resulted in substantial economic losses. Beyond natural disasters, accidents and public health crises also pose significant threats to life safety and economic stability. In the face of these catastrophic events, swift and effective emergency rescue operations are crucial to safeguarding public safety, maintaining social order, and minimizing losses [1]. It is imperative to develop and implement comprehensive preparedness, disaster mitigation, and response strategies to prevent casualties and alleviate the suffering in affected areas. Particularly, the prompt provision of medical services and other essential supplies can greatly ease the plight of victims and reduce the risk of loss of life [2]. This task presents a significant challenge. Firstly, in the aftermath of an emergency, measures like lockdowns and traffic restrictions often impede the access of large vehicles carrying essential rescue supplies. Secondly, the affected individuals at the rescue sites are more prone to irrational influences, making them particularly sensitive to delays in delivery, which can lead to a decrease in their satisfaction levels.

To address the challenge of rescue vehicles' difficulty in accessing rescue areas due to traffic control, a novel transportation model is introduced based on the Two-level Vehicle Routing Problem (2E-VRP). The concept of 2E-VRP was initially proposed by Perboli et al. [3] in 2011, presenting a mathematical model for 2E-VRP with a single warehouse. Since then, the rapid evolution of 2E-VRP has garnered increasing attention within the academic community. Soares et al. [4] extended the model by incorporating time windows, synchronization constraints, and multiple trips. Al Theeb et al. [5] delved into multiple commodity 2E-VRPs with time windows, taking into account customer-specific needs. Furthermore, various aspects of 2E-VRP have been explored, such as simultaneous pickup and delivery by the same vehicle [6], and time-dependent 2E-VRP, considering varying vehicle speeds [7].The 2E-VRP divides the transportation network into two tiers. The first tier (FE) comprises warehouses and satellites, typically located on the outskirts of cities or in suburbs, with roads mainly consisting of highways offering fast speeds and large capacities. The second tier (SE) consists of satellites and demand points, predominantly situated within cities and densely populated areas, characterized by slower vehicle speeds and smaller capacities, aligning with the scenario of damaged transportation facilities following a disaster.

However, the existing 2E-VRP literature primarily employs traditional two-level models, which, while saving time to some extent, have limitations. Specifically, they cannot predict in advance the optimal locations, quantities, and demands for each satellite, potentially leading to situations where demand exceeds time window constraints or cannot be met at certain demand points. To overcome this, a new two-level transportation approach is adopted. Initially, a transportation model for SE is constructed, assigning each demand point to a suitable satellite based on the demand point's time window and distance from the satellite. Subsequently, the required storage capacity for each satellite is determined, and FE is utilized to replenish cargo at the satellites. This method emphasizes coordination and cooperation between different tiers in a phased manner, effectively reducing the complexity of the problem.

In the process of distributing emergency supplies, vehicles are often susceptible to various interferences, such as road damage and weather changes. To address the decreased satisfaction of disaster-affected populations due to these irrational influences, it is crucial to consider the psychological expectations of customers in uncertain situations. To this end, prospect theory has been applied to develop more effective path planning solutions. Interference management, a methodology focused on real-time problem-solving in such scenarios, has garnered significant attention in international academic fields like management science, operations research, and systems engineering, emerging as a prominent research trend [8]. Ahmed [9] has demonstrated the ability to swiftly generate new solutions for faults encountered during vehicle transportation, with the algorithm's effectiveness being validated through rigorous experiments. Prospect theory, originally proposed by Kahneman [10], delves into how decision-makers choose outcomes in risky situations. When addressing vehicle interference issues, decision-making risks also arise. Consequently, scholars have leveraged prospect theory for in-depth exploration in this context. Zhou [11] investigated the impact of stress on decision-making behavior and developed a corresponding risk-taking model. Experimental results indicated an increased willingness to take risks under time pressure. By integrating prospect theory with the 2E-VRP model and considering vehicle disturbances, we can study the satisfaction of demand points concerning vehicle arrival times. This integration not only enhances the model's realism but also improves its accuracy and practicality.

Addressing complex NP-hard problems often proves challenging when relying solely on precise algorithms to obtain results. Consequently, this paper employs a hybrid heuristic algorithm to optimize and solve the Two-Echelon Vehicle Routing Problem with Time Windows (2E-VRPTW). A recurring strategy in 2E-VRP heuristic algorithms involves initially dividing the problem into two independent subproblems, with the second-tier VRP being tackled first [12]. Labarthe [13] implemented the Adaptive Large Neighborhood Search (ALNS) algorithm for 2E-VRP, introducing a novel destruction operator at the satellite level to facilitate exploration of diverse satellite configurations. Kahalimoghadam et al. [14] addressed a two-level distribution network involving parcel lockers, building upon the ALNS framework provided by Hemmelmayr et al. Subsequently, Wei et al. [15] extended this formulation by introducing a two-level vehicle routing problem that incorporates time windows, coverage options, and occasional drivers, employing both precise formulations and ALNS to solve it. Garside et al. [16] proposed an

Iterative Two-Stage Heuristic (ITSH) approach, which decomposes the problem into three subproblems. This method utilizes two Large Neighborhood Searches (LNS) to generate itineraries for first- and second-tier vehicles, while an Integer Programming (IP) model selects the optimal configuration. Building on this literature, we propose a hybrid genetic algorithm (HGA) that enhances the genetic algorithm (GA) by integrating it with the Large Neighborhood Search (LNS) algorithm to optimize 2E-VRPTW solutions. This approach not only retains the GA's advantages of speed and high-quality solutions but also mitigates the risk of converging to local optima.

The rest of this article proceeds as follows: In Section 2, we formulate a model tailored for the unique two-tier vehicle routing problem, seamlessly integrating soft time windows with prospect theory. Section 3 introduces a hybrid genetic algorithm, enhanced by the Large Neighborhood Search (LNS) method, to effectively solve the presented model. In Section 4, we undertake comprehensive numerical experiments, utilizing both hypothetical datasets and real-world case studies, to validate the model's feasibility and showcase the algorithm's enhanced performance. Lastly, in Section 5, we offer insightful humanistic perspectives and present the overarching conclusions of our study.

2 Problem description and model construction

2.1 Problem description

In the critical context of sudden emergency events, swift and precise resource allocation is of utmost importance. To address the challenge posed by large vehicles transporting rescue supplies being hampered by measures such as lockdowns and traffic controls following emergencies, this article conducts an in-depth exploration of the Two-Echelon Vehicle Routing Problem with Time Windows (2E-VRPTW).

Specifically, the distribution network is segmented into two distinct transportation networks. The first echelon (FE) comprises warehouses, multiple satellite stations, and a fleet of vehicles with substantial capacity constraints. The primary objective of the FE is to distribute materials to satellite stations, ensuring they maintain adequate stockpiles to satisfy downstream demand. The second echelon (SE) consists of a network of satellite stations, vehicles with more modest capacity limits, and numerous rescue points characterized by specific time windows and demand constraints. A visual representation of the transportation network is provided in Figure 1.

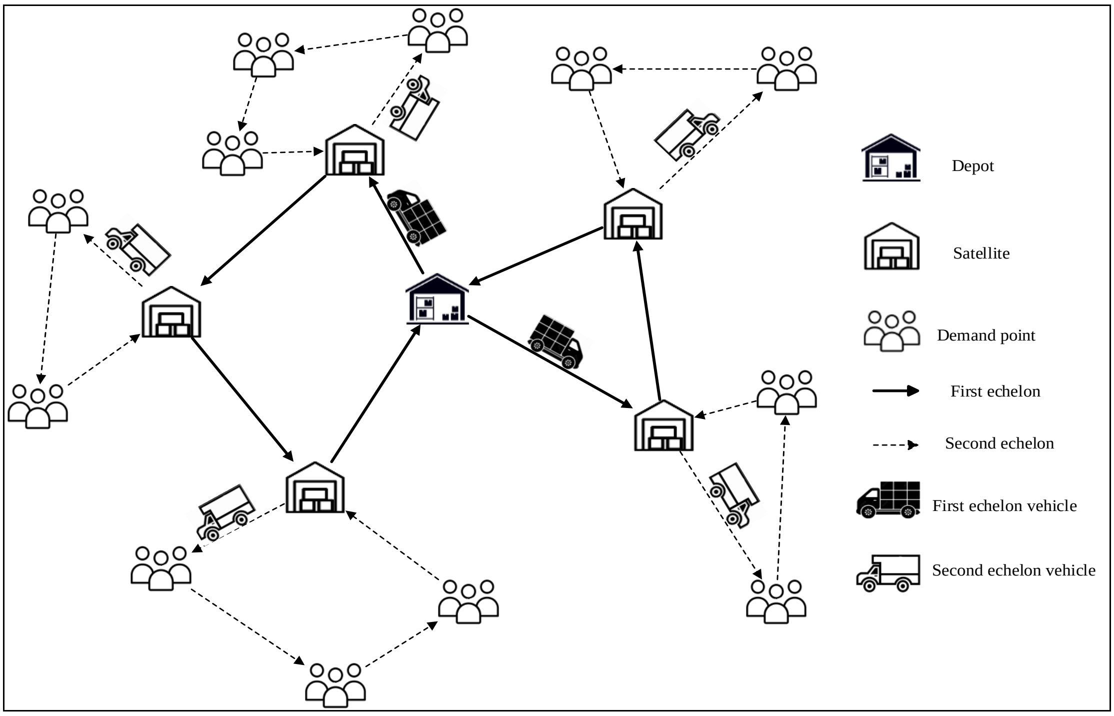

Figure 1. 2E-VRP Transportation Network Diagram

2.2 Parameter description and basic assumptions

Table 1. Symbol Explanation Table

Symbol	Meaning
O	Collection of warehouse nodes, $O = \{0\}$
S_i	Collection of satellite nodes, $S_i = \{A, B, C\}$
C	Collection of rescue point nodes, $C = \{1, \dots, c\}$
$[e_i, \ l_i]$	Delivery time window for rescue points
A_1	The first echelon rescue network node set, $A_1 = (O, S) = \{0,1, \dots, s\}$
A_2	The second echelon rescue network node set, $A_2 = (S, C) = \{1, \dots, s, s+1, \dots, s+c\}$
K_1	The first echelon vehicle group, $K_1 = \{1, \dots, k\}$
K_2	The second echelon vehicle group, $K_2 = \{1, \dots, l\}$
d_i	The demand for rescue point i, $i \in C$
D_i	The material reserve of satellite i, $i \in S$
q_1	The maximum capacity of the first echelon vehicle
q_2	The maximum capacity of the second echelon vehicle
c	Fuel cost per unit distance for vehicles
h	Unit vehicle usage cost
α	Tolerance level of rescue points
P_i	Time window penalty coefficient
s_i	Service duration of rescue point i
t_{ik}	The time when the vehicle leaves the satellite or arrives at the demand point,$i \in A_2, k \in K_2$
s_{ijk}	The distance traveled by the vehicle, $i \in A_1, j \in A_2, k \in (K_1 \cup K_2)$
u_{ik}	Record the location of the first level vehicle route, $i \in A_1, k \in K_1$
z_{ik}	Record the location of the second level vehicle route, $i \in A_2, k \in K_2$
x_{ijk}	Binary decision variable, if vehicle k travels from i to j, wait for 1; otherwise, wait for 0,$i \in O, j \in S, k \in K_1$
y_{ijk}	Binary decision variable, if vehicle k travels from i to j, wait for 1; otherwise, wait for 0, $i \in S, j \in C, k \in K_2$

The meanings represented by various symbols are shown in Table 1, and the following assumptions are made for 2E-VRPTW: 1) The two levels of transportation vehicles are of different types, and the speed and capacity of SE vehicles are greater than those of FE vehicles, driving at a constant speed without considering the influence of other factors; 2) The location, time window, cargo demand, warehouse and satellite center positions of each rescue point are known; 3) The types of goods between different customers are the same; 4) The warehouse or satellite returned by the vehicle is consistent with the warehouse or satellite from which it departed.

2.3 Model Building

In the complex and urgent context of emergency response, a mathematical model was carefully constructed with the core objective of minimizing the total cost of vehicle delivery for 2E-VRPTW. This model not only covers traditional vehicle fixed costs and driving costs, but also introduces two types of penalty costs: 1) penalties for violating time windows; 2) Punishment based on prospect theory.

2.3.1 Line Soft Time Window Penalty Function

In emergency rescue, time is life. Therefore, the model imposes strict penalty costs on vehicles that fail to deliver to the rescue site on time. Using a polyline soft time window penalty cost function, the function is shown in equation (1). This function adds a tolerable time window $[e_{i,} \ l_i]$ beyond the expected time window $[E_{i,} \ L_i]$ at the rescue point, where $E_i = e_i - \alpha s_i$、$L_i = l_i + \alpha s_i$, as shown in Figure 2. In this figure, the horizontal axis t represents time, and the vertical axis P_i represents the corresponding penalty amount.

$$p_i = \begin{cases} p_1(E_i - t_i) + p_2(e_i - E_i), & t_i < E_i \\ p_2(e_i - t_i), & E_i \le t_i < e_i \\ 0, & e_i \le t_i \le l_i \\ p_3(t_i - l_i), & l_i < t_i \le L_i \\ p_3(L_i - l_i) + p_4(t_i - L_i), & t_i > L_i \end{cases} \quad (1)$$

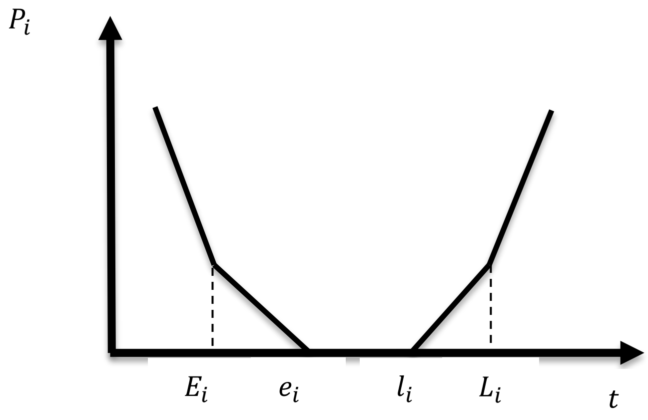

Figure 2. Penalty cost function under the soft time window with a broken line

2.3.2 Punishment function based on prospect theory

Considering the complexity and uncertainty faced by decision makers in emergency response, as well as the interference impact received during vehicle transportation, the model introduces prospect theory to evaluate the psychological costs incurred due to decision results deviating from expected goals. This penalty cost not only takes into account the material losses, but also covers the negative psychological effects caused by the decision results not meeting expectations, such as disappointment, anxiety, etc. The following is the value function of the prospect theory method, as shown in equation (2):

$$V^i(x) = \begin{cases} x^{\alpha^i}, x \ge 0 \\ -\lambda^i(-x)^{\beta^i}, x < 0 \end{cases} \quad (2)$$

$$i = 1, \dots, n$$

Among them, α^i,β^i and λ^i are parameters. Based on previous research on forward theory, $\beta = 0.88$ *and* $\lambda = 2.25$are taken.

According to the principles of prospect theory, in the early stages of the decision-making process, it is necessary to establish a suitable reference benchmark (with a value set to 0), and the state of profit or loss is judged relative to this benchmark, in order to determine whether the final result is profit or loss. In the delivery scenario, due to the uncertainty of the delivery time of goods, the arrival time when the vehicle is not disturbed is selected as the reference point. Considering that the delivery system mainly serves individuals with subjective consciousness, their perception of disturbances is often vague. Therefore, the value function was fuzzified and transformed into an dissatisfaction function to more accurately reflect the actual situation. The specific derivation process is shown in reference [17], as shown in equation (3):

$$u^i(x^i) = \begin{cases} 1, x^i \ge R^i \\ \lambda^i(x^i - O^i)^{\beta^i}, O^i \le x^i \le R^i \\ 0, 0 \le x^i \le O^i \end{cases} \quad (3)$$

$$i = 1, \dots, n$$

Among them, β^i and λ^i are parameters, $R^i = O^i + (1/\lambda^i)^{1/\beta^i}$。

The shape of the function is shown in Figure 3.

Because R^i is determined by β^i and λ^i, β^i and λ^i are different for different entities, and therefore R^i is also different.

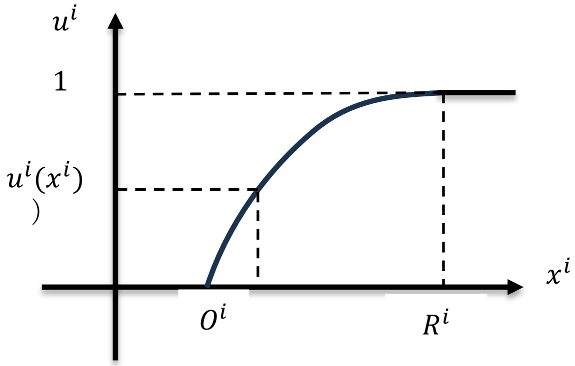

Figure 3. The unsatisfactory membership function of x^i

At this point, the arrival time of the vehicle must meet the time window of the demand point, and the dissatisfaction function is:

$$u(t_{jk}) = \begin{cases} 1, t_{jk} \geq R \\ \lambda(t_{jk} - t)^{\beta}, t \leq t_{jk} \leq R \\ 0, 0 \leq t_{jk} \leq t \end{cases} \quad (4)$$

Among them, β and λ are parameters, $R = t + (1/\lambda)^{1/\beta}$, and t is the planned arrival time of the vehicle.

2.3.3 Mathematical Model

In complex emergency scenarios, a staged mathematical model is proposed to efficiently and accurately plan the distribution path of rescue supplies. Given the urgency and uncertainty of emergency rescue, the model design fully considers the synergistic effect between SE and FE, as well as their unique distribution needs. Given the particularity of the model in this article, we will first construct a second level model and then plan the first level. Constraints (5) - (17) are SE constraints, and constraints (18) - (31) are FE path modeling. The model is as follows:

$$\min Z_1 = \sum_{k\in K_2} \sum_{(i,j)\in A_2} cs_{ijk} y_{ijk} + h \sum_{k\in K_2} \sum_{(i,j)\in A_2} y_{ijk} + \sum_{(i,j)\in A_1} y_{ijk} \left(p_i + u(t_{jk})\right) \quad (5)$$

$$\sum_{(i,j)\in A_2} y_{ijk} = \sum_{(i,j)\in A_2} y_{jik}, \forall k \in K_2, i \in S_i, j \in C \quad (6)$$

$$y_{ii'k} = 0, \forall k \in K_1, i, i' \in S_i \quad (7)$$

$$\sum_{(i,j)\in A_2} y_{ijk} = 1 \quad (8)$$

$$\sum_{i\in A_2} y_{ii'k} \leq 1, \forall i' \in C \quad (9)$$

$$z_{ik} - z_{jk} + |C| y_{ijk} \leq |C| - 1, \forall k \in K_2, (i,j) \in A_2 \quad (10)$$

$$q_2 y_{ijk} \leq \alpha * d_i \sum_{(i,j)\in A_2} y_{ijk}, \forall (i,j) \in A_2, k \in K_2 \quad (11)$$

$$t_{ik} = Lt * \sum_{(i,j)\in A_2} y_{ijk}, \forall i \in S_i, k \in K_2 \quad (12)$$

$$t_{ik} + t_{ijk} + s_i \leq t_{jk} + M * \left(1 - y_{ijk}\right), \forall (i,j) \in A_2, k \in K_2 \quad (13)$$

$$t_{jk} + t_{jik} \leq t_{ik} + M * \left(1 - y_{ijk}\right), \forall (i,j) \in A_2, k \in K_2 \quad (14)$$

$$t_{ik} \geq 0, i \in (S \cup C), k \in K_2 \quad (15)$$

$$z_{ck} \in Z^+, \forall c \in C, k \in K_2 \quad (16)$$

$$y_{ijk} \in \{0,1\}, \forall (i,j) \in A_2, k \in K_2 \quad (17)$$

Equation (5) is the objective function, representing the total of fixed vehicle costs, travel costs, and penalty costs. Constraint (6) is a traffic balance constraint for vehicles. Constraint (7) states that for each vehicle, it is not allowed to travel directly from the starting point to the destination. If no delivery task is assigned to the vehicle, no route will be generated. If the vehicle is assigned a task, it will serve at least one rescue point. Constraint (8) means that each vehicle can only depart from the warehouse once. Constraint (9) ensures that after each demand point, each vehicle can immediately process up to one demand point to transport resources. Constraint (10) ensures that the vehicle follows the one-way non round trips rule, therefore it is necessary to add and eliminate loop constraints. Constraint (11) states that once a vehicle is assigned to serve a certain demand point, it should at least meet a certain proportion of the demand. Constraint (12) provides the departure time of the vehicle from the satellite. Constraint (13) states that the arrival time of each vehicle at the rescue point shall not be earlier than the arrival time at the satellite plus the unloading time at the satellite and the intermediate travel time to the demand point. Constraint (14) states that the arrival time of each vehicle returning to the satellite from the rescue point shall not be earlier than the time of arrival at the rescue point plus the intermediate travel time of each time period. Constraints (15) - (17) define the variable domain.

$$\min Z_2 = \sum_{k\in K_1}\sum_{(i,j)\in A_1} cs_{ijk}x_{ijk} + h\sum_{k\in K_1}\sum_{(i,j)\in A_1} x_{ijk} \quad (18)$$
$$\sum_{(i,j)\in A_1}\sum_{k\in K_1} q_1\, x_{ijk} = \sum_{(i,j)\in A_2}\sum_{k\in K_2} q_2 y_{ijk} \quad (19)$$
$$\sum_{i\in S_i} D_i = \sum_{(i,j)\in A_2}\sum_{k\in S_i} q_2 y_{ijk} \quad (20)$$
$$\sum_{(i,j)\in A_1} x_{ijk} = \sum_{(i,j)\in A_2} x_{jik}, \forall k\in K_1 \quad (21)$$
$$x_{ii'k} = 0, \forall k\in K_1, i, i' \in O \quad (22)$$
$$\sum_{(i,j)\in A_1} x_{ijk} \le 1, \forall k\in K_1 \quad (23)$$
$$\sum_{i\in A_1} x_{ii'k} \le 1, \forall i' \in S \quad (24)$$
$$u_{ik} - u_{jk} + |S|x_{ijk} \le |S| - 1, \forall k\in K_1, (i,j)\in A_1 \quad (25)$$
$$t_{ik} = Lt * \sum_{(i,j)\in A_1} x_{ijk}, \forall i\in O, k\in K_1 \quad (26)$$
$$t_{jk} + t_{jik} \le t_{ik} + M * \left(1 - y_{ijk}\right), \forall (i,j)\in A_1, k\in K_1 \quad (27)$$
$$t_{ik} \ge 0, i\in (O\cup S_i), k\in K_1 \quad (28)$$
$$u_{sk} \in Z^+, \forall s\in S_i, k\in K_1 \quad (29)$$
$$x_{ijk} \in \{0,1\}, \forall (i,j)\in A_1, k\in K_1 \quad (30)$$

Equation (18) is the objective function, representing the total of fixed vehicle costs, travel costs, and penalty costs. Constraint (19) connects the first and second tiers and specifies that the total flow from the warehouse to the satellite is equal to the total demand provided by the satellite. Constraint (20) indicates that the reserve of each satellite is determined by the second tier. Constraint (21) is a traffic balance constraint for vehicles. Constraint (22) states that for each vehicle, it is not allowed to travel directly from the starting point to the destination. Constraint (23) means that each vehicle can only depart from the warehouse once. Constraint (24) ensures that after each demand point, each vehicle can immediately process up to one demand point to transport resources. Constraint (25) ensures that the vehicle follows the one-way non round trip rule, therefore it is necessary to add an elimination loop constraint. Constraint (26) provides the departure time of the vehicle from the warehouse. Constraint (28) states that the arrival time of each vehicle returning to the warehouse from the satellite shall not be earlier than the time of arrival at the satellite plus the intermediate travel time of each time period. Constraints (28) - (30) define the variable domain.

3 Hybrid Algorithm Design

Given that the Two-Echelon Vehicle Routing Problem with Time Windows (2E-VRPTW) is an NP-hard problem, solving it with complex and precise methods can be challenging, especially considering the urgent context in which it arises. Consequently, this paper employs heuristic algorithms to tackle this challenge. The optimization of rescue material distribution paths encompasses numerous variables and constraints, including distribution distance, time window limitations, vehicle cargo capacity, and individual satisfaction. These intricate factors render it difficult for traditional optimization algorithms to identify the global optimal solution within a constrained timeframe.

Genetic Algorithms (GA), with their unique encoding strategy, fitness evaluation, selection, crossover, and mutation operations, are adept at efficiently searching for optimal solutions within the solution space, thereby enhancing both the efficiency and quality of the solution. To address the multi-objective optimization model of 2E-VRPTW, this paper acknowledges the limitations of standard GA, such as its dependence on initial solutions, slow convergence speed, and susceptibility to converging to local optima. To mitigate these issues, the paper introduces an enhanced GA that incorporates a hybrid algorithm with Large Neighborhood Search (LNS) to solve the problem. The algorithm's pseudocode is presented in Table 2, and the specific improvements are outlined as follows:

3.1 Improved Getic Algorithm

3.1.1 Chromosomal coding system

An enhanced encoding method is introduced, building upon the traditional one-dimensional integer encoding approach [18]. In this refined method, chromosomes exclude codes for satellites and warehouses. Instead, the coding sequence of chromosomes is determined by the order of access to demand points. For instance, C= {2,1,3} signifies that demand point 2 is prioritized for access, followed by point 1, and lastly point 3. This innovative encoding technique enables the algorithm to concentrate more on optimizing the sequence of access among demand points during the solution space search, rather than on the combination of warehouses and demand points. Consequently, this contributes to a faster convergence rate and enhanced search efficiency.

3.1.2 Construct initial population

Using integer encoding strategy to encode chromosomes, and carefully constructing the initial chromosome population with the help of insertion heuristic algorithm. The specific steps are as follows:

Step 1: Set k as the number of vehicles that can be deployed by all satellites, and po as the expected population size. Randomly select an unmarked disaster point i (ranging from 1 to n) and mark the demand point i;

Step 2: Initialize all vehicles in the vehicle set K (loading capacity, current running time, whether the vehicles are marked);

Step 3: Randomly select an unmarked vehicle k_1 from a satellite in the vehicle set K, and label the vehicle;

Step 4: Create an empty set N and place all the earliest demand points with the upper limit of the time window into set N. Randomly select a demand point from set N and insert it into the path of k_1, while also inserting it into the chromosome sequence;

Step 5: Subsequently, greedy insertion is used to check all demand points one by one and insert them into the vehicle with the minimum increase in the objective function, while satisfying the time window and demand constraints;

Step 6: If the chromosome length is less than n, that is, there are demand points that cannot meet the time window or demand constraints, place these demand points into the set Nr, randomly select demand points from the set Nr, insert them into the vehicle with the minimum increase in the objective function, without satisfying the time window or demand constraints, until the chromosome length reaches n;

Repeat the above operation until the number of chromosomes generated reaches po, at which point the initial chromosome population is generated.

3.1.3 Selection, crossover, and mutation operators

For parental selection, the roulette wheel method is employed, followed by the generation of new chromosomes through a sequence-based mixed crossover and simple exchange mutations. The crossover method utilized is an 'Order-based Hybrid Crossover,' which blends the attributes of Partially Matched Crossover (PMX) and Ordered Crossover (OX). The specific crossover steps are detailed as follows:

Step 1: Randomly select a crossover point location that divides the parental chromosome into two parts;

Step 2: Copy the first half of parent 1 (from the starting point to the intersection) directly into offspring 1, and mark these genes as used;

Step 3: Select genes from parent 2 that have not yet appeared in offspring 1 and fill them in the remaining positions of offspring 1 in order;

Step 4: Copy in reverse order from the latter half of parent 2 (from the crossover point to the end) to offspring 2 (skipping genes that have already appeared in offspring 1);

Step 5: Select the remaining unused genes from parent 1 and fill them into offspring 2. When all genes are replicated into offspring without duplication, the crossover process is complete. The cross diagram is shown in Figure 4.

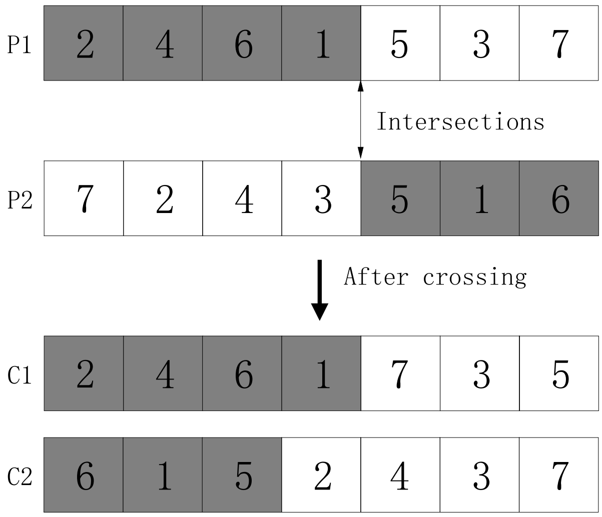

Figure 4. Cross diagram

3.1.4 Repair operator

Following the crossover and mutation processes, the coding sequence of the chromosomes undergoes modifications, necessitating the re-planning of vehicle routes. In this context, the greedy insertion heuristic is employed to systematically examine each chromosome node and insert it into the vehicle that results in the minimal increase in the objective function value, while simultaneously adhering to the time window and demand constraints.

3.2 Large Neighborhood Search

To address the challenge of genetic algorithms frequently converging to local optima, this paper incorporates Large Neighborhood Search (LNS) as a strategy to escape local loops. Specifically, LNS is applied to the chromosomes after the genetic algorithm completes its crossover, mutation, and repair processes.

LNS is a powerful metaheuristic algorithm that revolves around the core idea of exploring extensive neighborhoods within the solution space. This is achieved by initially "destroying" the current solution and subsequently undertaking a "repair" process, with the aim of discovering solutions of superior quality. During program execution, the performance of each deletion and insertion operator is carefully monitored, enabling dynamic adjustments to their respective weights. Consequently, the selection probabilities of these operators are continuously updated.

To ensure a more effective attainment of global optimization, a variety of methods are employed for the deletion and insertion operations. The detailed solution process unfolds as follows:

3.2.1 Remove operation

In the context of Large Neighborhood Search (LNS), the removal operation plays an integral role within the destroy operator, and its efficacy has a direct impact on the overall performance of the algorithm. Consequently, the design of effective removal strategies is of paramount importance in LNS. To this end, three distinct types of destructive operators were employed, as detailed below:

Shaw operator: Following Shaw's suggestion [19], extract a set of related demand points. In this study, the relationship between demand points i and j is quantified using $RC\ (i,j)$ as follows: $RC(i,j) = \varphi_1 D_{ij} + \varphi_2(|l_i - l_j| + |u_i - u_j|) + \varphi_3(|d_i - d_j|) + \varphi_4 sr_{ij}$

Correlation quantification includes four aspects: distance, time window, demand, and route. The weights of each of them are denoted as $\varphi_1, \varphi_2, \varphi_3$ and φ_4, respectively. If i and j belong to the same route, let $sr_{ij} = -1$. $RC\ (i,j) \in [0, \varphi_1 + \varphi_2 + \varphi_3 + \varphi_4]$. The smaller the $RC(i,j)$, the closer the relationship between demand points.

Firstly, randomly select a demand point, remove it, and then calculate the correlation between the removed demand point and other demand points. Finally, based on the ascending order of correlation values, remove the demand points with the highest correlation, and repeat this process until the number of deleted points reaches the predetermined number of demand points.

Worst removal operator: The main purpose of the operator is to minimize the total distance traveled. Select a demand point from the customer set, calculate and save the saving value of the path distance before and after the point is deleted. Repeat this process for the remaining demand points to obtain the saving value of the path distance before and after all demand points are deleted. Sort the saving values of all demand point distances in descending order, and select the top m customer points with larger saving values to delete from the corresponding path.

Random removal: This operator randomly selects m customers from the set of demand points and removes them from the corresponding paths to increase the diversity of the search.

3.2.2 Repair operation

The primary objective of the repair operation is to reintegrate the elements that were removed during the destruction phase back into the remaining solution structure. This is accomplished through various insertion strategies, which generate new candidate solutions. The expectation is that these new solutions will yield improvements in the objective function value, thereby progressively converging towards the global optimal solution.

Greedy insertion operator: inserts the removed demand points one by one into the remaining solutions to minimize the increase in objective function values (such as total cost, total distance, etc.). This method is usually more efficient in generating high-quality candidate solutions, as it tends to select the positions that can bring the maximum immediate benefit for insertion.

Rule based Greedy insertion operator: By calculating, find the increase in the objective function value of the optimal insertion position for all removed demand points (such as the position with the minimum increase in total cost, total distance, etc.), insert the demand point with the minimum increase value, and then repeat the calculation until all demand points are inserted.

Table 2. Pseudo code of Hybrid Genetic Algorithm

Algorithm 1 Hybrid Genetic Algorithm for 2E-VRPTW
Procedure GENETICALGORITHM
Initialize population and related structures
Generate initial population
for each generation **do**
Create new population
for each pair of parents in population **do**
if crossover condition met **then**
Perform crossover to generate offspring
else
Select parents directly as offspring

```
        end if
    end for
    for each offspring in new population do
if mutation condition met then
Apply mutation
        end if
        Repair and evaluate offspring
        if periodic condition met then
         Perform large neighborhood search
        end if
    end for
    Update population with best individuals from new
    end for
  Post-process best solution
    Initialize second tier population based on best solution
end procedure
```

4 Case analysis

Using the Sichuan earthquake as a case study, let's consider a scenario where there are 20 disaster-affected areas, one emergency medical supplies reserve center, and three satellites established between the rescue points and the reserve center. At the second level, each vehicle has a loading capacity of 5 tons, incurs a usage cost of 100 yuan, and travels at a speed of 60 km/h. At the first level, vehicles have a loading capacity of 10 tons, a usage cost of 150 yuan, and a speed of 45 km/h. Following a disturbance, the vehicle speed is reduced to 30 km/h, with a unit transportation cost of 3 yuan per kilometer. The service time at each rescue point is proportional to the demand for supplies. The relevant parameters of HGA algorithm are set as follows: population size $m = 50$, maximum iteration times $gen\_max = 200$, crossover probability $\alpha = 0.85$, mutation probability $\beta = 0.01$.

The above data is input into the algorithm model of this article and run 10 times to obtain the optimal result. The objective function is 1760.64 yuan, the total driving distance is 352.61km, and the average dissatisfaction rate is 14.05%. The driving path is shown in Table 3, and the driving path diagram is shown in Figure 5.

All experiments in this article were simulated on a desktop computer with Intel (R) Core (TM) i5-9300HF CPU @ 2.40GHZ and 16GB of memory. Visual Studio 2019 was used for programming and solving, as follows:

Table 3. Driving Path Table

Echelon	Cost	Number of vehicles	Vehicle number	Distribution paths
SE	1330.89	4	1 2 3 4	A-5-9-15-2-12-A A-3-11-17-18-6-A B-1-7-13-10-16-B C-19-20-8-14-4-C
FE	429.75	2	1 2	0-C-B-A-0 0-A-0

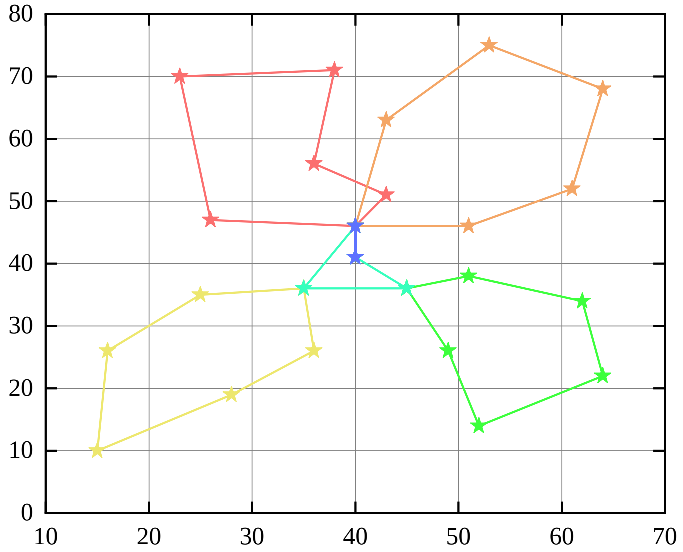

Figure 5. Driving Path Diagram

4.1 Comparative analysis

Select data from the Solomon dataset for improvement, which includes six different types of data. Among them, the C-class dataset represents clustered customers, the R-class dataset represents randomly generated customers, and the RC class dataset is a mixed dataset that contains both C-class and R-class features. Three simulation experiments will be conducted this time:

(1) We used the C101 case study to test the effectiveness of the HGA algorithm, and compared it with the optimal paths obtained through heuristic algorithms in the history of various case studies provided by Solomon Benchmark's website to verify the feasibility of our algorithm.

(2) Using unimproved GA algorithm, hyper heuristic genetic algorithm [20], and the HGA algorithm proposed in this paper to solve the improved R101 case, compare and analyze the optimal solutions of each algorithm, and evaluate the improvement efficiency of the algorithm proposed in this paper.

(3) In scenarios targeting two different customer sizes (50, 100), a case study was selected from each category and improved using an improved hybrid genetic algorithm (HGA) combined with the 2E-VRP model proposed in this paper and the traditional VRP model of classical genetic algorithm (GA). The results of both methods were compared and analyzed to evaluate the feasibility of the proposed model.

The relevant parameters of HGA algorithm are set as follows: population size $m = 80$, maximum iteration times $gen\_max = 200$, crossover probability $\alpha = 0.85$, mutation probability $\beta = 0.01$, The relevant parameters of GA algorithm are consistent with HGA. The tolerance level of demand point $\alpha = 0.5$, unit transportation cost $c = 8$, unit vehicle usage cost $h = 60$, penalty coefficients $p_1 = 1$, $p_2 = 0.5$, $p_3 = 1.5$, and $p_4 = 2$ for the curved soft time window, and penalty coefficients$p_1 = p_2 = 0.5$ and $p_3 = p_4 = 1.5$ for the ordinary soft time window.

(1) Comparative analysis of HGA algorithm for solving the optimal solution of Solomon case and the published optimal solution

Conduct experiments using the C101 example in the Solomon dataset and calculate the driving distance for demand points of 50 and 100, respectively. Using the HGA algorithm for solving, calculate 10 times for each case and record the optimal solution. The experimental results are shown in Table 4. From the data in Table 4, it can be seen that in the case of 25 customer points, the difference from the known optimal solution is within 0 About 27%; In the case of 50 customer points, the optimal solution of our algorithm is superior to the known optimal solution; In the case of 100 customer points, the difference from the known optimal solution is 0 About 2%. From this, it can be seen that the HGA proposed in this article is feasible.

Table 4: Comparison of Historical Heuristic Optimal Paths and the Optimal Solution of Our Algorithm

Numerical example	Optimal path	Optimal solution of algorithm
C101.25	191.3	191.81
C101.50	362.4	363.24
C101.100	827.3	828.94

(2) Comparative analysis of optimal solution results of GA algorithm, hyper heuristic GA and HGA algorithm

Table 5 shows the optimal results of the unimproved GA algorithm, hyper heuristic genetic algorithm, and the HGA algorithm proposed in this paper for solving case R101. At this time, all three algorithms use a polyline soft time window. The optimal path diagrams obtained by each algorithm are shown in Figure 6.

Table 5 Optimal Solution and Delivery Route

Algorithm	Cost	Vehicle number	Vehicle loading rate	Distribution paths
GA	5951.12	1	57.5	0-13-15-23-21-2-22-4-0
		2	20.0	0-14-17-8-24-25-0
		3	41.5	0-12-1-10-20-9-3-0
		4	24.5	0-6-18-7-19-11-0
		5	22.5	0-5-16-0
Super heuristic GA	4763.80	1	47.0	0-5-16-6-18-8-17-13-0
		2	53.5	0-14-15-2-22-23-4-25-0
		3	28.0	0-21-12-3-24-1-0
		4	37.5	0-11-19-7-10-20-9-0
HGA	4361.53	1	53.0	0-18-8-7-19-11-10-20-9-1-0
		2	59.0	0-12-21-22-23-4-25-24-3-0
		3	54.0	0-6-5-17-16-14-15-2-13-0

From the data in Table 5, it can be seen that the HGA algorithm has an optimization rate of 26.71% compared to the GA algorithm, and an optimization rate of 8.44% compared to the hyper heuristic GA. In addition, the optimal solution of the HGA algorithm in this paper uses fewer vehicles than the other two algorithms, and the corresponding vehicle loading rate of each vehicle is above 50%, greatly improving the utilization of resources. Prove that the algorithm in this article has good improvement effect.

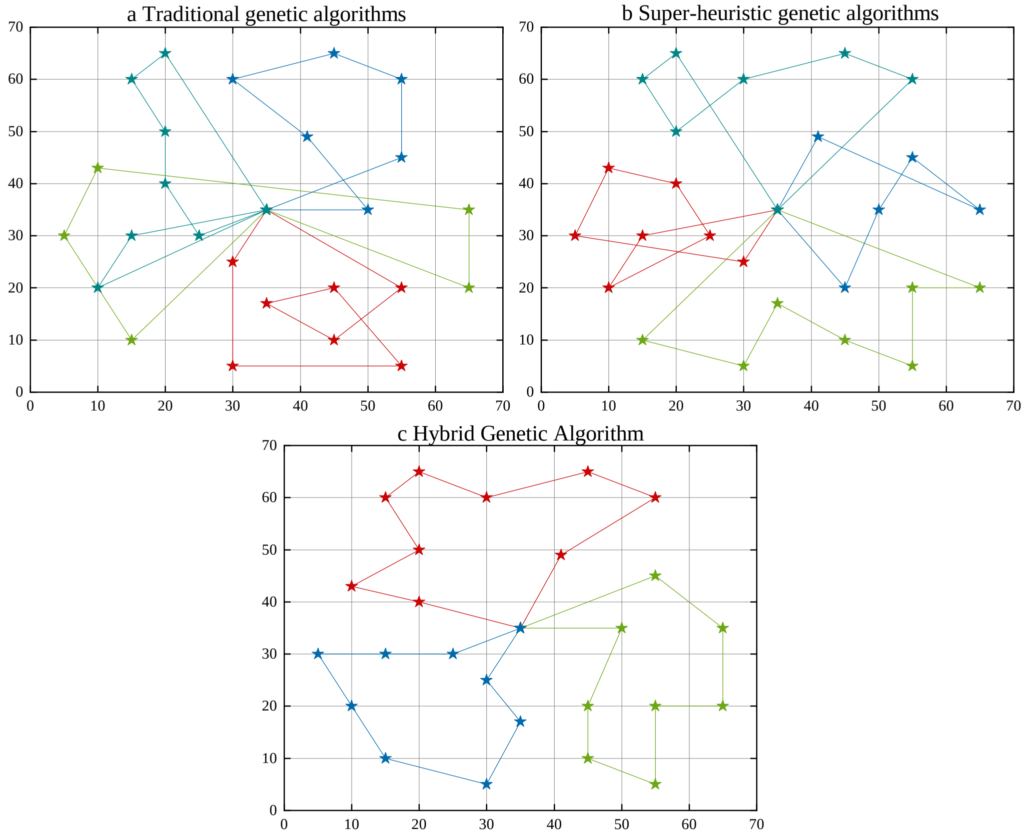

Figure 6. Optimal Path Diagram for Each Algorithm

(3) Comparison of GA algorithm solving VRP and HGA solving 2E-VRP results

Run the two algorithms independently with their corresponding models 10 times, and take the optimal solution (C_{best}) and average value (C_{ave}) of the target value as the running results. Gap_{best} represents the percentage difference between the optimal solutions of two algorithms, which is the optimization rate of HGA algorithm compared to GA algorithm.

$Gap_{best} = \frac{C_{best}(\text{HGA}) - C_{best}(GA)}{C_{best}(GA)} \times 100\%$. The experimental results are shown in Tables 6-7.

Table 6. Solution Results for a Case Study with a Demand Point Scale of 50

Data set	Traditional soft time window+ Traditional genetic algorithm+ VRPTW		Line Soft Time Window+ Hybrid Genetic Algorithm+2E-VRPTW		Gap_{best}/%
	Number of vehicles	Cost	Number of vehicles	Cost	
C101	8	9022.34	5	4231.8	-53.1
R101	8	10280.99	7	8722.03	-15.2
RC101	7	9162.42	5	6812.92	-25.64
C201	4	23250.96	3	19090.39	-17.89
R201	5	14186.39	3	13795.64	-2.75
RC201	5	13415.4	3	12013.91	-10.45

From the comparison of the solution results for the case with a demand point quantity of 50 in Table 6, it can be observed that all cases have negative Gap_{best} values. This indicates that under this demand point scale, the optimal solution obtained by HGA algorithm is superior to GA algorithm. Especially in the solution of example C101, the HGA algorithm showed a significant optimization effect, with an optimization rate exceeding 50%, and the number of vehicles required by the HGA algorithm was consistently less than that of the GA algorithm. These results indicate that the HGA algorithm has stronger solving ability when dealing with cases with medium demand point sizes.

Table 7. Results of solving a case study with a demand point scale of 100

Data set	Traditional soft time window+ Traditional genetic algorithm+ VRPTW		Line Soft Time Window+ Hybrid Genetic Algorithm+2E-VRPTW		Gap_{best}
	Number of vehicles	Cost	Number of vehicles	Cost	
C101	16	25596.96	10	8166.89	-68.09
R101	13	20208.62	12	16446.88	-18.61
RC101	14	22836.91	13	15525.31	-32.02
C201	7	44594.26	3	5542.56	-87.57
R201	6	24840.39	4	18899.54	-23.92
RC201	10	28964.64	4	19731.52	-31.88

From the comparison of the solution results for the case with a demand point quantity of 100 in Table 7, it can be observed that all cases have negative Gap_{best} values. This indicates that under this demand point scale, the optimal solution obtained by HGA algorithm is superior to GA algorithm. Especially in the solving of examples C101 and C201, the HGA algorithm demonstrated extremely significant optimization effects, with an optimization rate exceeding 50%, and the number of vehicles required by the HGA algorithm was consistently less than that of the GA algorithm. These results indicate that the HGA algorithm has stronger solving ability when dealing with cases with medium demand point sizes.

Figure 7 shows a comparison of the convergence curves of the optimal solutions obtained by two algorithms for example R101 with two different customer sizes. It can be seen from the figure that the HGA algorithm has a significantly faster search speed for feasible solutions in the early stage and can escape from local optima through local search operations.

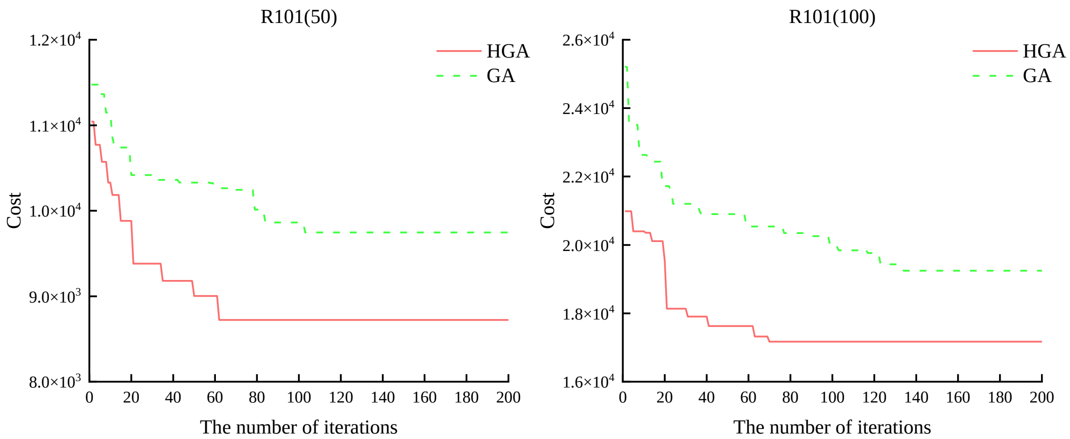

Figure 7. Comparison of the convergence curves of the optimal solution obtained from solving example R101 using GA and HGA

Conclusion

This article examines the material distribution challenge during sudden emergencies. Initially, a mixed model integrating prospect theory and a penalty model was formulated to consider individual biases and delivery satisfaction. Given traditional models' limitations, a novel two-tier model with a soft time window for polyline routes was adopted. This boosts vehicle efficiency, minimizes dissatisfaction, and improves solution quality. The effectiveness of a hybrid genetic algorithm, incorporating large neighborhood search, was validated using the Solomon test set.

Acknowledgements This research was supported by National Natural Science Foundation of China(72401128), Basic Science (Natural Science) Research Project of Higher Education Institutions of Jiangsu Province (23KJB62006), the Humanities and Social Sciences Program of the Ministry of Education (23YJCZH201), Jiangsu Social Science Foundation (23GLLC015) and the National Social Science Foundation of China (21CJY055).

Declarations

Competing Interests The authors declare that they have no financial and personal relationships with other people or organizations that can inappropriately influence our work. There is no professional or other personal interest of any nature or kind in any product, service and/or company that could be construed as influencing the position presented in, or the review of, the manuscript entitled.

Ethical and Informed Consent This paper does not contain any studies with animals performed by any of the authors. Informed consent was obtained from all individual participants included in the study.

Reference

1. Weng X, She W, Fan H, Zhang J, Yun L, Multi-depot vehicle routing problem with drones in emergency logistics. Cluster Computing, 2025, 28(1): 1-27.
2. Chang, K. H., Chiang, Y. C., Chang, T. Y., Simultaneous location and vehicle fleet sizing of relief goods distribution centers and vehicle routing for post-disaster logistics. Computers & Operations Research, 2024, 161:106404.
3. Perboli, Guido, Roberto Tadei, et al., The two-echelon capacitated vehicle routing problem: Models and math-based heuristics. Transportation Science, 2011, 45(3): 364-380.
4. Soares R, Marques A, Amorim P, Parragh SN., Synchronisation in vehicle routing: classification schema, modelling framework and literature review. European Journal of Operational Research, 2024, 313(3): 817-40.
5. Al Theeb N, Abu-Aleqa M, Diabat A., Multi-objective optimization of two-echelon vehicle routing problem: Vaccines distribution as a case study. Computers & Industrial Engineering, 2024, 187: 109590.
6. Lehmann J, Winkenbach M., A matheuristic for the Two-Echelon Multi-Trip Vehicle Routing Problem with mixed pickup and delivery demand and time windows. Transportation Research Part C: Emerging Technologies, 2024, 160: 104522.
7. Gutierrez A, Labadie N, Prins C., A Two-echelon Vehicle Routing Problem with time-dependent travel times in the city logistics context. EURO Journal on Transportation and Logistics, 2024, 13: 100133.
8. Mohamed NB, Hassan MZ, Kaddoum G., RSMA-enabled interference management for Industrial Internet of Things networks with finite blocklength coding and hard-ware impairments. IEEE Transactions on Machine Learning in Communications and Net-working, 2024, 2: 1319-1340.
9. Ahmed F, Huynh N, Ferrell W, Badyal V, Padmanabhan B., Vehicle re-routing under disruption in cross-dock network with time constraints. Expert Systems with Appli-cations, 2024, 237: 121517.
10. Ning T, Jin H., A cloud based improved method for multi-objective flexible job-shop scheduling problem. Journal of Intelligent & Fuzzy Systems, 2018, 35(1): 823-829.
11. Zhou YB, Ruan SJ, Zhang K, Bao Q, Liu HZ., Time pressure effects on deci-sion-making in intertemporal loss scenarios: an eye-tracking study. Frontiers in Psychology, 2024, 15: 1451674.
12. Sluijk N, Florio A M, Kinable J, et al., Two-echelon vehicle routing problems: A literature review. European Journal of Operational Research, 2023, 304(3): 865-886.
13. Labarthe O, Ahmadi G, Klibi W, Deschamps JC, Montreuil B., A sustainable on-demand urban delivery service enabled by synchromodality and synergy in passenger and freight mobility. Transportation Research Part C: Emerging Technologies, 2024, 161: 104544.
14. Kahalimoghadam M, Thompson RG, Rajabifard A., Determining the number and location of micro-consolidation centres as a solution to growing e-commerce demand. Journal of Transport Geography, 2024, 117: 103875.
15. Wei J, Xu X, Yang B., Data-driven vehicle

rental and routing optimization: An application in online retailing. Computers & Industrial Engineering, 2024, 197: 110588.
16. Garside AK, Ahmad R, Muhtazaruddin MN., A Recent Review of Solution Approaches for Green Vehicle Routing Problem and its variants. Operations Research Perspectives, 2024, 100303.
17. NING Tao, WANG Xu-ping, HU Xiang-pei, et al., Disruption management optimal scheduling for logistics distribution based on prospect theory. Control and Decision, 2018, 33(11): 2064-2068.
18. Wang S, Zhu X, Shang P, Lin X, Yang L, Tavasszy L., Two-echelon multi-commodity multimodal vehicle routing problem considering user heterogeneity in city logistics. Expert Systems with Applications, 20 24, 252: 124141.
19. Jiang J, Dai Y, Yang F, Ma Z., A multi-visit flexible-docking vehicle routing problem with drones for simultaneous pickup and delivery services. European Journal of Operational Research, 2024, 312(1): 125-37.
20. HAN Yajuan, PENG Yunfang, WEI Hang, et al., Hyper-heuristic genetic algorithm for vehicle routing problem with soft time windows. Computer Integrated Manufact uring Systems, 2019, 25(10): 2571-2579.