scientific reports

OPEN

Rich vehicle routing optimization based on variable neighborhood descent and differential evolution algorithm

Haifei Zhang[1]✉, Yuzhou Zhang[1], Fen Zhao[1], Lujie Zhou[2] & Bailing Zhou[1]

In order to reflect the vehicle routing problem more realistically, meet the planning needs of different decision makers for vehicle routing, seek multiple equivalent optimal paths, and improve the diversity of the optimal solution set, we regard Rich Vehicle Routing Problem (RVRP, which also means vehicle routing problem with multi constraints) as a multi-modal multi-objective optimization problem. This paper considers the RVRP under four constraints, which are more practical, such as complex road network constraint, load constraints, time window constraint and demand splitting constraint. In addition, when solving this problem, We have designed a method that combines Differential Evolution (DE) algorithm with Variable Neighborhood Descent (VND) algorithm. Firstly, in order to expand the search range of the population, an Oppositional Learning (OL) mechanism is introduced in the basic DE to broaden the search range of solutions. Secondly, in response to the problem of premature convergence and falling into local optima in the basic differential evolution algorithm, a VND local search method is embedded to enhance the population search capability. By optimizing the mathematical model and improving the solving algorithm, the performance of the proposed method was evaluated on the standard benchmark instance of the problem. The experimental results showed that the constructed model and the improved solving algorithm can solve the problem of multiple equivalent optimal paths in logistics distribution. This method achieved the best comprehensive performance and was superior to the most advanced RVRP solving method. It has great potential in practical engineering.

Keywords Logistics distribution, Multi constraint problem, Vehicle routing problem, Multi-modal multi-objective optimization, Differential evolution algorithm

In the past decades, vehicle routing problem (VRP) and its variants have been widely popularized because they can simulate the practical applications in various fields. Their applications include transportation planning, supply chain management in logistics network, production management, etc. The goal of VRP is to design a group of optimal distribution routes for vehicles of a certain scale, so as to provide services for customers in logistics distribution. It represents the essence of vehicle allocation and route planning under the lowest cost in logistics distribution. Therefore, it is a key problem in logistics distribution and one of the most widely studied problems in the field of combinatorial optimization. Since the truck scheduling problem proposed by Dantzig and Ramser[1], researchers have been studying the relationship between vehicle routing planning and delivery planning. It is considered as a typical case of VRP, involving the distribution of goods from central depots to geographically dispersed customers. Due to the influence of many factors such as transportation enterprises, customers and external environment, the current vehicle routing planning of logistics distribution is facing severe challenges.

This paper studies an important variant of VRP, namely, the reality oriented multi constraint VRP (Rich VRP, RVRP). The RVRP extends classical VRP by incorporating multiple real-world constraints and objectives, making it a more practical yet complex variant. In this study, RVRP is defined by four key dimensions of richness: Complex Road Network Constraints: Vehicles must navigate urban road networks with traffic restrictions (e.g., one-way streets, no-entry zones) and dynamically optimized shortest paths. Heterogeneous Fleet and Capacity

[1]College of information engineering of Nanjing, XiaoZhuang University, Nanjing 211171, Jiangsu, China. [2]School of Automobile and Transportation, Tianjin University of Technology and Education, Tianjin 300222, China. ✉email: 18215199587@163.com

Constraints: Vehicles have fixed capacities, and demand splitting is allowed to maximize load utilization. Time Window Constraints: Customers impose strict delivery time intervals, with penalties for early/late arrivals. Demand Splitting Constraints: The demand of a single customer node can be distributed by multiple vehicles in multiple times. Unlike prior RVRP formulations, this model uniquely integrates road network complexity with demand splitting and multi-objective trade-offs, reflecting realistic urban logistics challenges.

VRP and its basic variants have been widely discussed in the literature. RVRP for practical problems has become a new research trend in recent years. Paola Pellegrini et al.[2] studied the RVRP with four constraints: multiple time windows, heterogeneous fleet, maximum duration, and multiple visits. D Pisinger et al.[3] studied the RVRP with five constraints: time window constraint, capacity constraint, multi depots, site-dependent, the open VRP, and simultaneous pickup and delivery. Goel A et al.[4] studied the RVRP of time window constraint, vehicle heterogeneity constraints, multi-dimensional capacity constraint, order/vehicle compatibility constraints, simultaneous pick-up and delivery, multi depots and other constraints. Subramanian et al.[5] studied the RVRP with capacity constraint, asymmetric constraint, open, simultaneous pickup and delivery, mixed pickup and delivery, multi depots and multi depot with mixed pickup and delivery. Subsequently, two review studies on RVRP came into being. Arias et al.[6] conducted a comprehensive research and summary on RVRP. Rahma Lahyani et al.[7] classified and defined RVRP, summarized the composition of the problem, constraint definition and solution method. After that, Qi et al.[8] studied the RVRP of multi station, multi time window, multi journey and multi vehicle types. Rabbouch et al.[9] studied the RVRP of multi warehouse heterogeneous finite fleets (vehicle quantity constraints, vehicle capacity constraint, time window constraint, heterogeneous fleets, different vehicles) with time windows.

RVRP is produced to meet the actual needs of transportation. As a NP-hard problem, it is also a multi-objective optimization problem. The importance of its objective function varies from field to field. For example, for the food distribution and medical industries, delay time is critical. The freight transport industry can consider the total journey as the key objective to minimize compared with other objectives, because the fuel consumption is proportional to the driving distance. Therefore, from an economic point of view, it is important to minimize the total distance traveled by all vehicles. For small industries, the minimization of the number of vehicles may be the highest priority compared to other goals. When planning the vehicle path, the decision-makers hope to obtain multiple paths that meet the target requirements at the same time, so as to ensure the stability of the decision, that is, there are at least two equivalent global Pareto optimal solutions corresponding to the same point on Pareto front (PF)[10,11]. That is to say, when we are solving the multi-objective optimization problem RVRP, the vehicle path that satisfies the constraint conditions is an optimal path set. In order to find more equivalent optimal paths corresponding to the same objective optimal solution[12], RVRP can be regarded as a multi-modal multi-objective optimization problem (MMOP).

In recent years, researchers have proposed many multi-modal multi-objective optimization algorithms (MMEAs) to solve MMOP. MMOP has multiple Pareto solution sets, which are usually crowded when mapped to the Pareto front in the target space, even corresponding to the same Pareto front. Therefore, when designing MMEAs, it is usually necessary to consider both decision space and target space. Based on this, many MMEAs with good performance are proposed[13–15]. They can simultaneously obtain multiple equivalent global optimal solutions in the problem, providing more choices for decision makers. Li et al.[13] proposed a multi-modal multi-objective optimization algorithm called MMEAWI, which is based on weighted index. It fuses the diversity information of solutions in the decision space into an objective spatial performance index to maintain the diversity of the decision space. In addition, the algorithm introduces the convergence archive to ensure more effective approach to Pareto frontier. Ming and Gong[16] proposed a coevolutionary algorithm called CMMO. The algorithm uses coevolution, target relaxation technology, specially designed environment and mating selection to balance the convergence and diversity of target space and decision space, so as to solve MMOPs more effectively. Li et al.[17] proposed an algorithm called HREA, which uses hierarchical ranking method to rank individuals in the population according to different levels to promote the selection and evolution of different solutions in the population. The algorithm also uses a local convergence quality evaluation method to better maintain the diversity of decision space.

MMOP aims to locate multiple equivalent Pareto-optimal solutions corresponding to the same objective values, which is critical for decision-makers seeking diverse yet equally optimal paths in logistics planning. However, despite the notable advancements in the theoretical framework and algorithm design of MMOP, its application in real-world logistics scenarios, such as RVRP, continues to confront significant challenges. The multi-objective optimization problem lacks a single, definitive global optimal solution, and the abundance of non-dominated solutions cannot be readily implemented in practice. Consequently, the pursuit of solutions must focus on identifying an equivalent set of globally optimal solutions.

Although many multi-objective optimization algorithms have been proposed to solve RVRP and its related variants[18–20], due to the complexity of problem modeling, the difficulty of solving, and the multi-modality of the problem, the research results are relatively few. Paola Pellegrini et al.[2] used ant colony optimization algorithm to solve RVRP considering four constraints. D Pisinger et al.[3] proposed a hybrid heuristic algorithm to solve the problem. Goel A et al.[4] proposed an iterative method to change the neighborhood structure in the search process. Subramanian et al.[5] proposed a hybrid algorithm to solve the RVRP considering seven different constraints, and solved a series of set partitioning (SP) models by using a mixed integer programming (MIP) solver. Srivastava et al.[21] proposed a non-dominated sorting genetic algorithm (NSGA-II) with target specific mutation operator. Konstantakopoulos et al.[22] proposed a multi-objective evolutionary algorithm (MOEA) with improved construction algorithm and crossover operator. Sethanan et al.[23] proposed a hybrid differential evolution algorithm with fuzzy logic controller genetic operator. Peng et al.[24] proposed a hybrid evolutionary algorithm combined with variable neighborhood search. The above researches show that evolutionary algorithm has certain advantages in solving RVRP and related problems.

In recent years, hybrid metaheuristic methods have shown significant advantages in complex optimization problems, effectively improving search efficiency and solution quality through multi technology fusion. In the field of vehicle path optimization, Sathyamurthy et al.[25] innovatively combined the perturbation mechanism of simulated annealing (SA) with the crossover mutation of genetic algorithm (GA), and embedded a mixed integer linear programming (MILP) model to solve the multi warehouse rechargeable vehicle path problem. Dynamic balance was achieved through the local search ability of SA and the global exploration property of GA. Similarly, the Ferreira team[26] proposed a variable neighborhood search algorithm for green vehicle routing and two-dimensional loading constraints, integrating lower bound programs, open space heuristics, and constraint planning models to systematically address the coupling problem of loading feasibility verification and path optimization. In the field of computational intelligence, Wu et al.[27] used the RankNet surrogate model to predict individual ranking relationships and combined it with the Local Estimation Distribution Algorithm (EDA) to construct a hybrid optimization framework, significantly improving the search efficiency of high-dimensional coverage problems. In the field of healthcare, Narasimhan et al.[28] used genetic algorithms to optimize random forest feature subsets, synchronously solving dynamic demand allocation and disease prediction problems, achieving collaborative optimization of feature selection and model performance. These studies all demonstrate the core advantages of hybrid metaheuristic methods: breaking through the limitations of a single algorithm through complementary techniques such as global local search balance, machine learning embedding, and constraint modeling, providing systematic solutions for complex problems in multi constraint, high-dimensional, and dynamic scenarios.

According to the analysis of previous studies, although the research on RVRP has achieved some research results, there are still some problems: (1) The current research on RVRP does not consider the impact of urban complex road network on vehicle routing, they both ignore that the logistics distribution process is based on the complex urban road network. (2) The objective of RVRP is relatively single. At present, the research on RVRP mainly solves the objective from a certain angle, such as minimum total cost, minimum travel time, minimum the average waiting time of customers, minimum travel time, etc. However, in the actual logistics distribution process, multi-objective optimization needs to be considered. (3) The resource utilization rate is not high. For most logistics enterprises or distribution centers, the number of vehicles and distribution personnel available for each transportation task are limited. Therefore, how to reduce the logistics transportation cost, maximize the use of limited resources, and improve the loading rate of transportation vehicles is very important for the related research of route optimization.

Urban logistics distribution faces escalating demands for cost efficiency, environmental sustainability, and customer satisfaction, yet traditional VRP models often oversimplify constraints like road networks and single-objective optimization, failing to address real-world complexities. To bridge this gap, this study proposes the RVRP framework that integrates four critical constraints—complex road networks, vehicle capacity, time windows, and demand splitting—through a comprehensive analysis of practical logistics challenges, constructing an integer programming model aligned with multi-constraint routing requirements and actual road network conditions. Formulated as a multi-modal problem with a Pareto front reflecting trade-offs between fuel costs, delivery times, and resource utilization, RVRP demands algorithms capable of escaping local optima while preserving solution diversity. Addressing this combinatorial complexity, the hybrid OL-DEVND algorithm innovatively combines Differential Evolution (DE) and Variable Neighborhood Descent (VND): it embeds Opposition-Based Learning (OL) to expand the search space during initialization, ensuring coverage of dispersed Pareto regions, while VND's adaptive neighborhood switching refines equivalent routes without sacrificing diversity. This dual mechanism synergizes DE's global exploration with VND's local precision, overcoming modality loss in classical DE-based MMOP solvers and outperforming existing methods in route equivalence preservation. The framework ensures robust convergence to diverse Pareto-optimal routes through adaptive constraint handling, enhancing solution quality for multi-objective logistics planning.

The main contributions of this paper are as follows:

1) Comprehensive RVRP Modeling: Unlike traditional models, which address isolated constraints (e.g., time windows or multi-depots), this model integrates four critical dimensions into a unified integer programming framework. This aligns with real-world logistics operations where these constraints coexist.*.
2) Oppositional Learning-Enhanced DE: While DE-based methods are known for global search, they often overlook population diversity in combinatorial spaces. The OL is introduced during initialization to generate adversarial solutions.
3) VND-Embedded Local Search: Existing MMOP algorithms rely on fixed mutation operators, limiting their ability to escape local optima in RVRP. The VND is embedded into the adaptive neighborhood exchange to improve the quality of the solution.

Problem description and model establishment

Problem description

Combined with the actual logistics distribution situation of most logistics enterprises in the market, in the actual logistics distribution process, due to the limitations of the complex urban road network, under the condition of ensuring the maximum vehicle capacity, considering that the customer demand can be split, and most customers have specified the time interval for order distribution, under these constraints, the problem can be defined as a rich vehicle routing problem (RVRP), which can be described as:

For the distribution center in a known area, there are several vehicles sent from the distribution center. The distribution requirements of multiple logistics orders are completed orderly and without repetition in the complex urban road network. If the customer's demand exceeds the vehicle carrying capacity, the demand can be split. These orders limit the time window of specific distribution. If the distribution vehicles are earlier or later than this time window, then a certain time penalty cost should be added to the final total transportation cost. Under the above constraints, the logistics distribution lines should be reasonably planned to minimize the number of vehicles required, the total distribution distance, the distribution time and the distribution cost.

Model assumptions

The problem of urban logistics distribution path planning is related to many factors. The mathematical model is very complex and has many constraints. In order to facilitate modeling, the following assumptions are made in this study:

(1) Only consider the logistics distribution of a single logistics distribution center.
(2) The vehicles responsible for logistics distribution must take the distribution center as the starting point and return to the distribution center after completing all customer order distribution tasks.
(3) Each vehicle only completes the distribution of one line.
(4) The demand and location coordinates of each customer are known and fixed.
(5) The arc formed between customer nodes is an un-directed arc. For example, when an un-directed arc is formed between a customer node *i* and a customer node *j*, it means that the distribution vehicles can be transferred from customer node *i* to customer node *j*, or from customer node *j* to customer node *i*.
(6) The arc formed between customer nodes has two-way weight, which represents distance and time cost.
(7) In the process of vehicle distribution, the impact caused by temporary vehicle failure or wrong goods distribution will not be considered for the time being.
(8) Time Window Constraints: Soft constraints with penalty costs. Early/late arrivals are permitted but penalized proportionally to deviation time.
(9) Fleet Homogeneity: All vehicles have identical capacity and operational costs.
(10) Demand Splitting: Customer demand can be split across multiple vehicles if $q_i > w$, ensuring full resource utilization.

Notation definition

The notations and meanings related to the model are represented in Table 1.

Establishment of objective function

The multi constraint vehicle routing problem is based on the actual logistics distribution. In the actual logistics distribution, the logistics distribution system is composed of multi constraint conditions such as vehicle capacity constraint, urban complex road network, customer time window constraint, customer demand split constraint, etc. when the complex multi constraint logistics distribution problem is optimized, the following mixed integer programming mathematical model is established:

f_1 : the number of vehicles required to complete the distribution task:

$$\min\ f_1 = R = \left\lceil \sum_{i=1}^{n} \frac{q_i}{w} \right\rceil \tag{1}$$

f_2 : the total driving distance of the vehicles, taking the minimum value :

$$\min\ f_2 = \min\ D = \min\ \sum_{i,j\in C}\sum_{r\in R} d_{ij}x_{ij}^r \tag{2}$$

f_3 : the total vehicle delivery time:

$$\min\ f_3 = \min(\sum_{i,j\in C}\sum_{r\in R}(\frac{q_j d_{ij} x_{ij}^r}{v} + Td_j) + \beta\sum_{i\in C}\sum_{r\in R} Z_i^r \cdot \max\{b_{ir} - LT_i, 0\}) \tag{3}$$

f_4 : the total cost of completing logistics distribution, which is composed of driving cost, vehicle fixed cost and time delay cost, and takes the minimum value:

$$\min\ f_4 = \min(\sum_{r\in R}\sum_{i,j\in C} f y_{ij} x_{ij}^r + \sum_{r\in R} G \sum_{j\in C'} x_{oj}^r + l) \tag{4}$$

$$l = \alpha(\sum_{r\in R}\sum_{i\in S^r}\sqrt{(x_i - \frac{\sum_{i\in S^r} x_i}{|S^r|})^2 + (y_i - \frac{\sum_{i\in S^r} y_i}{|S^r|})^2}) \tag{5}$$

l is the time delay cost, and its value is proportional to the sum of the distances from all customers in the line to the center of the geographical location. Here, the time delay cost is only used to compare the advantages and disadvantages of the schemes, and the value of a single scheme has no actual operational significance.

The four objectives are inherently interrelated, reflecting real-world logistics trade-offs:

Minimizing the number of vehicles (f_1) often requires consolidating deliveries into fewer routes, which increases individual route lengths and total distance (f_2). For example, reducing vehicles from 10 to 8 may

Notation	Meaning	Notation	Meaning
α	time delay cost corresponding to unit distance	W	maximum carrying capacity of vehicle
$V=(C,A)$	road network and customers involved in logistics distribution	R	number of vehicles required to complete the distribution task
n	number of customers	(x_i,y_i)	coordinates of customer i
$C'=C/\{C_0\}$	set of n customers	$C=\{0,1,2,\ldots,n\}$	collection of distribution centers and customers
S^r	the customer set served in route r, that is, the customer set that the r vehicle is responsible for distribution.	A	the set of paths consists of the shortest path between any two points in C. the traffic fault paths that must be avoided in A are put into setNC,$NC\in A$
C_0	Distribution Centre	q_i	the demand of customer i
$\lceil\rceil$	round up to	G	fixed cost of vehicles
β	time cost coefficient caused by violating the delivery time specified by the customer	$\lvert S^r\rvert$	the number of elements contained in the collection, that is, the number of customers
FY	cost matrix, cost$fy_{ij}\in FY$corresponding to each path$a_{ij}\in A$	LT_i	the latest time that the distribution vehicle is allowed to arrive at the customer.
v	speed of distribution vehicles	Td_j	waiting time at customer j
b_{ir}	actual time when vehicle r arrives at customer point i	d_{ij}	distribution distance from point i to point j

Table 1. Notation definition.

extend average route distances. This conflict arises from the fixed vehicle capacity *w*, forcing longer detours to serve all customers. Shorter routes (f_2) reduce fuel costs but risk violating time windows (f_3), incurring penalties. Conversely, prioritizing strict time compliance may require additional vehicles or routes, raising operational costs (f_4).Timely deliveries (f_3) reduce penalty costs embedded in f_4. For instance, eliminating a 1-hour delay for a customer with β=$10/hour directly saves $10 in f_4.f_4 aggregates fixed vehicle costs, fuel expenses, and penalties, making it a composite metric influenced by f_1, f_2, and f_3. Optimizing f_4 inherently balances other objectives but may obscure specific trade-offs.

Constraints:

Each customer is visited at least once:

$$\sum_{r\in R}\sum_{i\in C} x_{ij}^{r} \geqslant 1 \tag{6}$$

A decision variable, if and only if the vehicle in the r^{th} route passes through the arc (i,j), $x_{ij}^{r}=1$, otherwise $x_{ij}^{r}=1$.

$$x_{ij}^{r} \in \{0,1\}\,, r\in R, i, j \in C' \tag{7}$$

The distribution needs of each customer are met.

$$\sum_{r\in R} y_i^r = q_i, \forall i \in C' \tag{8}$$

The distribution demand of customer *i* in the r^{th} line.

$$y_i^r, i\in C, r\in R, q_i \geqslant y_i^r \geqslant 0 \tag{9}$$

The conservation of flow, that is, the number of vehicles entering a point is equal to the number of vehicles leaving the point.

$$\sum_{i\in C} x_{ip}^{r} - \sum_{j\in C'} x_{pj}^{r} = 0 \; p \in C' \tag{10}$$

The traffic obstacles that must be avoided do not appear in the distribution line.

$$\sum_{r\in r} x_{ij}^{r} = 0, a_{ij} \in NC \tag{11}$$

The number of arc edges between served customers in each line is equal to the number of served customers minus 1.

$$\sum_{i\in S^r}\sum_{j\in S^r} x_{ij}^{r} = |S^r| - 1, r\in R, S^r \subseteq C' \tag{12}$$

The customer *i* can only be served when the vehicles passe by.

$$\sum_{j\in C} x_{ij}^{r} q_i \geqslant y_i^r \tag{13}$$

Whether the distribution task has time requirements.

$$Z_i^r \in \{0,1\} \;\forall r \in R, i \in C \tag{14}$$

The distribution volume of each distribution vehicle does not exceed the upper limit of vehicle capacity. In urban logistics, vehicles often operate near full capacity to minimize trips and fuel costs. For example, a vehicle with *w* = 500 kg serving a customer with q_i= 600 kg must split the demand into two trips (500 kg + 100 kg), directly impacting route planning and costs.

$$\sum_{i\in C} q_i y_{ir} \leqslant w \quad \forall r \in R, i \in C \tag{15}$$

Whether the vehicle passes customer *i*.

$$y_{ir} \in \{0,1\} \quad \forall r \in R, i \in C \tag{16}$$

Each logistics distribution task has vehicle distribution.

$$x_{ij}^{r} \leqslant y_{ir} \quad \forall r \in R, i \in C, j \in C \tag{17}$$

The departure time of the vehicle is 0.

$$b_{or} = 0\,,\ \forall r \in R \tag{18}$$

The time iteration relationship of the delivery vehicle arriving at the customer.

$$b_{jr} = x_{ij}^{r}(b_{ir} + \frac{d_{ij}}{v})\ \forall r \in R, i \in C, j \in C \tag{19}$$

The dwell time of distribution to each customer, and the dwell time has a certain linear relationship with the distribution volume.

$$Td_j = f(q_j)\ \forall j \in C \tag{20}$$

The customer distribution demand is greater than 0.

$$b_{jr} - b_{ir} \geqslant 0\ \forall r \in R, i \in C, j \in C \tag{21}$$

Soft time windows reflect practical scenarios where minor delays are tolerable but costly. A customer requiring delivery between 10:00–12:00 may accept a 12:30 arrival with a penalty of β=$10/hour, balancing service quality and operational flexibility.

Solving method

In recent years, there have been many researches on the methods of solving the optimal vehicle routing. The commonly used methods mainly include exact algorithm and heuristics[29,30]. According to the analysis in *Sect. 1*, RVRP is a multi-modal and multi-objective optimization problem. When solving this kind of problem, NSGA-II[31] and differential evolution algorithm[32] have shown good results.

DE, as a new type of intelligent algorithm, has a simple principle, few controlled parameters, good robustness, and is easy to implement. Its essence is a multi-objective (continuous variable) optimization algorithm, mainly used to solve the overall optimal solution in multidimensional space. Due to its simple structure and ability to effectively enhance population diversity, the DE algorithm may fall into local optima during the evolution process. Therefore, this paper introduces the VND[33] in DE to avoid falling into local optima. In addition, in order to obtain more effective solutions, an adversarial learning mechanism is introduced in the initialization process to expand the population.

Multi objective optimization framework

Addressing the multi-objective and multimodal nature of RVRP, the OL-DEVND algorithm accomplishes multi-objective optimization via the following steps:

Initialization A diversified initial population is generated leveraging Oppositional Learning.

Global exploration Differential Evolution (DE) is employed to generate new solutions, encompassing both the objective space and decision space.

Local refinement Variable Neighborhood Descent (VND) enhances the quality of solutions to circumvent local optima.

Solution set update Non-dominated solutions are screened using a combination of Pareto dominance and crowding distance.

Termination condition The Pareto front is outputted upon meeting convergence criteria or reaching the maximum iteration count.

The flowchart of the OL-DEVND hybrid algorithm is shown in Fig. 1.

The pseudocode of the OL-DEVND hybrid algorithm is shown in Algorithm 1.

Algorithm 1: Algorithm OL-DEVND

```
Input: Population size N, maximum iteration times T, road network constraints
Output: Pareto optimal solution set
Initialization: Generate the current population and the opposing population, merge them, and select Top-N individuals
for t = 1 to T do
Mutation and crossover: DE/rand/1 strategy generates offspring
Local search: VND for offspring applications (Exchange → Insert → 2-opt)
Merge populations: parent + offspring
Non dominated sorting: Hierarchical screening of Pareto frontiers
Crowding distance calculation: Sort solutions on the same layer by diversity
Choose a new generation population: retain the first N individuals
end for
Return Pareto optimal solution set
```

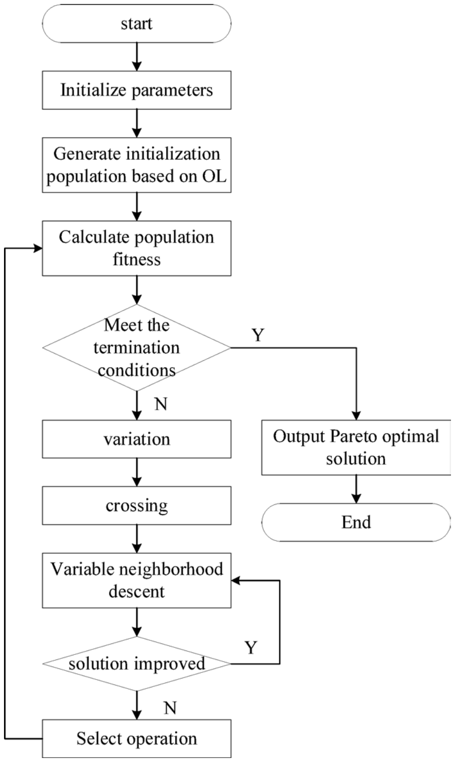

Fig. 1. Flowchart of the DEVND.

Population initialization based on oppositional learning

OL is a commonly used strategy for escaping from local optimal solution positions[34]. OL not only helps individuals quickly escape from their current location, but also increases the likelihood of their fitness values being better compared to their current worst position.

Generate opposing solutions through the strategy of adversarial learning and the position of the current solution, as shown in Eq. (22):

$$X_{opposite} = rand \times (ub + ul) - X \tag{22}$$

Where, X represents the current position of the solution.

ub and represent the upper and lower bounds of the problem solution, respectively. $rand$ is a random number in [0,1]. In addition, the diversity of the generated space increased by random numbers and the variability of the current position enhances the unpredictability of the exploration process. In exploring mechanisms, this unpredictability is crucial.

Definition Opposite Point Assuming there exists a number x on [1, u], the opposite point of x is defined as $x' = 1 + u - x$. Extending the definition of opposition to space, let $p = (x_1, x_2, ..., x_d)$ be a point in a d -dimensional space, where $x_i \in [l_i, u_i]$, $i = 1, 2, ..., d$, and its opposition is $p' = (x'_1, x'_2, ..., x'_d)$, where $x'_i = l_i + u_i - x_i$.

According to the above definition, the specific steps for generating the initial group using the adversarial learning strategy are shown in Algorithm 2.

Algorithm 2: Initialization Method Based on Oppositional Learning

Set population size N
for i = 1 to N do
for j = 1 to d do$X_i^j = l_i^j + rand(0,1) \cdot (u_i^j - l_i^j)$
end for
end for
or i = 1 to N do
for j = 1 to d do$OX_i^j = l_i^j + u_i^j - X_i^j)$
end for
end for
Merge$\{X(N) \cup OX(N)\}$, select Nindividuals with the best fitness values as the initial population

DEVND algorithm design

Encoding operation

There are various forms of encoding operations for vehicle path planning, one of which is to represent all customer points with numbers, form a feasible solution after algorithm optimization, and randomly set fixed points based on this feasible solution. Here, the fixed point is set as the central vehicle factory in the encoding operation, usually using a fixed number "1" or "0". This feasible solution is divided into several paths, and each vehicle path starts from the central vehicle factory and returns to it. This type of coding has a drawback of easily ignoring vehicle capacity. When encountering models with limited vehicle capacity, it is difficult to effectively meet the demand. The second type is the coding method, which adopts the "sort first, cluster later" method. Chromosome coding is a non repeating sorting that includes all customer point numbers. Path segmentation meets the requirement that the total cargo demand does not exceed the vehicle load. That is, in one chromosome, all customer point numbers are arranged in a line from left to right. This coding method occupies relatively small storage space and the scheme will cover all distribution points. The coding efficiency is high. Assuming the optimal chromosome distribution route is 4 → 7 → 3 → 8 → 9 → 6 → 10 → 2 → 5, the vehicle capacity is… When delivering to customer point "8", the fully loaded cargo volume C is reset to zero. At this time, the vehicle needs to return to the central depot, The path information of the first car is 4 → 7 → 3 → 8.

DEVAD algorithm steps

The reciprocal of the objective function for solving the model is set as the fitness function F, and then the differential evolution algorithm is selected, including the crossover and mutation operators.

(1) Selection operator.

To rank all feasible solutions in the population, it is necessary to choose the one with the highest fitness function and the lowest objective function. There are five selection functions for the selection operator, and the roulette wheel strategy is used for selection. The calculated probability is:

$$P(x_i) = \frac{F(x_i)}{\sum_{j=1}^{N} F(x_j)} \tag{23}$$

$$LQ_i = \sum_{j=1}^{N} P(x_j) \tag{24}$$

Among them, $P(x_i)$ is the probability of each feasible solution being inherited into the next generation population, and LQ_i is the cumulative probability of each individual.

(2) Crossover operator.

The initial generation of random integers r_1 and r_2 within the [0,1] interval determines the intersection position between the offspring and the parent, and crosses the intermediate data between the two positions. Defined binomial crossover with CR = 0.8 and ensured at least one dimension from V_i is retained.

(3) Mutation operation.

The mutation strategy involves randomly selecting two points and swapping their positions, using the "DE/rand/1" strategy with a scaling factor of F = 0.5.

(4) Reinsert offspring.

The re insertion strategy is to replenish the mutated individuals back into the population after crossover, in order to obtain the optimal solution in this iteration.

(5) Change neighborhood descent search to update solutions.

The meaning of neighborhood in different problems is also different. It is a relatively mature improved local search algorithm. The main idea of this algorithm is to use multiple different neighborhoods for systematic search. When the current neighborhood cannot improve the solution, it switches to another neighborhood to improve the quality of the solution. The current neighborhood searches for improved solution quality and continues to search in this neighborhood. The variable neighborhood descent search algorithm is embedded into the genetic algorithm. Each iteration of the genetic algorithm will generate a new individual, and the VND algorithm is used to locally search for the individual path.

The neighborhood search algorithm sets N neighborhood structures, with the neighborhood structure being $N_k = N_1, N_2, ..., N_n$. Here, N is set as the Exchange optimization neighborhood, Insert optimization neighborhood, and 2-opt optimization neighborhood. A vehicle path is selected as the initial neighborhood, and optimization starts from the Exchange optimization neighborhood. The mutual switching of optimization neighborhoods is called neighborhood action. If an improved solution is found in this neighborhood, the disturbance continues in this neighborhood. If no improved solution is found, the operation is repeated in the next neighborhood. In this study, the path optimization method is used. The search effect and range of the three types of path optimization neighborhoods are the same [1,1], all randomly selecting a vehicle path and performing neighborhood operations within this vehicle path, as shown in Figs. 2, 3 and 4.

Exchange optimization: Choose a path, exchange two positions in the path, and finally form a new path.

Insert optimization: Select a path and insert a node into another location along the path.

2-opt optimization: Select a path, traverse each node in the path in reverse order, and generate a new path.

(6) Multi objective solution set update strategy.

In the multi-objective solution set update strategy, the algorithm ensures the convergence and diversity of the solution set through the following mechanism: firstly, non dominated sorting is used to perform Pareto

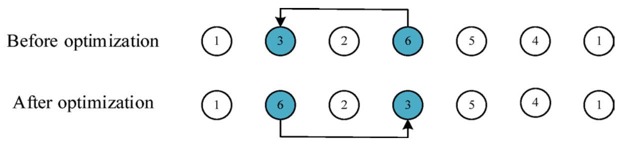

Fig. 2. Exchange optimization.

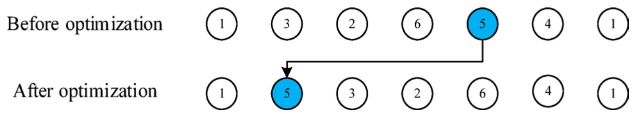

Fig. 3. Insert optimization.

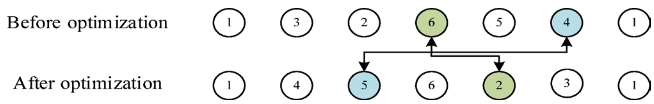

Fig. 4. 2-opt optimization.

stratification on the population, and individuals are divided into different levels based on the target value, with priority given to retaining non dominated solutions located at the Pareto front, thus selecting the globally optimal candidate solution set. Secondly, for individuals within the same Pareto layer, the distribution density of evaluation solutions is calculated through crowding distance: the weighted sum of Euclidean distances between adjacent individuals in the target space is calculated (weights reflect the importance of each target), and sparse solutions are retained to maintain the diversity of the solution set and avoid local clustering. Finally, the elite retention strategy is introduced to merge the parent and child populations, and then apply non dominated sorting and crowding distance calculation in sequence to select a new generation population that combines high quality and high diversity. This process effectively balances the global exploration and local development capabilities, ensuring that the algorithm approaches the true Pareto front and covers its diverse regions simultaneously in multi-objective optimization.

Experimental results and analysis
Experimental settings
Parameter settings To select the optimal parameter configuration, this paper conducted preliminary experimental tuning to evaluate the impact of these parameters on the algorithm performance, and also referred to the existing literature. The final parameter settings are as follows.

The DE parameter is set to a population Popsize of 50, a maximum iteration number IterMax of 100, a crossover probability Pc of 0.8, and a mutation probability Pm of 0.1. The feasible solution of the VRP problem is first encoded and an initial population is generated. Based on the input conditions of the mathematical model, the fitness of the initial population is calculated and the population is selected. The termination condition according to the objective function is: (1) the current optimal solution remains unchanged for 10 consecutive generations; (2) When the number of iteration steps exceeds 100.

Hardware configuration To verify the effectiveness of the proposed method, MATLAB language was used for experimental simulation. The computer was configured with Intel Core i7-3630QM 2.40 GHz, 8GB RAM, and executed on Windows 10 system.

Data set The proposed method was evaluated on Augerat Set-P[35], a widely recognized CVRP benchmark, and Zhou & Wang's real-world logistics dataset[36], which includes 45 instances with varying customer sizes (50–250 nodes), time windows, and vehicle capacities. Augerat Set-P provides standardized instances for reproducibility, while Zhou & Wang's dataset reflects urban logistics challenges like traffic constraints and split deliveries. No new instances were proposed, as our focus was on enhancing algorithmic performance under established benchmarks.

Evaluation indicators A single performance indicator cannot comprehensively measure the performance of the multi-objective optimization algorithm. Therefore, we use four metrics, that is, inverse generational distance (IGD)[37], coverage metric (C-metric)[38], 1/HV (HV is hyper volume[37]) and 1/PSP. PSP is Pareto sets proximity (PSP)[32]. PSP reflects the overlap rate and distance between real PS and obtained PS. IGD and c-metric are the most commonly used performance indicators for multi-objective optimization problems. 1/HV and 1/PSP can measure performance in decision space and target space respectively. The smaller the value, the better the performance. They are common evaluation indicators for MMOP.

Comparison algorithm To verify the effectiveness of the proposed algorithm, this paper compares HDMMODE[10], INSGA_II[39], ACO[40], ADPRA[41], Sim-BRIG-LS[42] for the problem. All algorithms are executed under identical conditions, which means using the same starting and ending criteria, the same number of starting search points, the same dataset, and the same hardware for running the algorithm. To reduce the influence of randomness, all experiments were carried out 30 times.

Results analysis of augerat Set-P

To better prove the efficiency of the algorithm proposed in this paper, this paper randomly selects instances of different sizes and types on the Augerat Set-P dataset. Due to the fact that this dataset is a standard dataset rather than actual running data, and this dataset is a CVRP dataset with no time window data in the it, the effectiveness of the proposed algorithm is explored using f_1 and f_2 as the measurement criteria. In the table, the units of the two objective function values are number of vehicles, m, respectively.

From Table 2, it can be seen that on standard CVRP instances with fixed vehicle capacity and customer demand, the hybrid algorithm proposed in this paper exhibits strong advantages compared to other algorithms in calculating the two objectives of vehicle number and driving distance. Taking P-n40-k5 as an example, OL-DEVND reduces the number of vehicles by 1 and shortens the driving distance by about 2.52% compared to the optimal Sim BRIG-LS algorithm among other comparison algorithms.

To analyze the significance of the differences in the experimental results, the experimental results of all algorithms were T tested by randomly selecting examples of different sizes and types. The test results are shown in Table 3. It can be seen from Table 3: Based on the T-test results, all the functions are significantly different than each of the other methods.

Results analysis of real-world logistics dataset

Due to Zhou & Wang's real-world logistics dataset reflects urban logistics challenges like traffic constraints and split deliveries. This paper conducted a dual validation of the objective function and measurement indicators on this dataset.

Table 4 shows the comparison results of all algorithms on this dataset under different customer scale and time window constraints. In the table, the first column names the instance in the form of "$num_1 - num_2 - num_3$", where, num_1 represents the number of customers, num_2 represents the index of different types of vehicle capacity, and num_3 represents the index of time window configuration, the units of the four objective functions are number of vehicles, m, min and $.

To analyze the significance of the differences in the experimental results, the experimental results of all algorithms were T tested by randomly selecting examples of different sizes and types. The test results are shown in Table 5. It can be seen from Table 5: Based on the T-test results, all the functions are significantly different than each of the other methods.

Tables 6 and Table 7 show the comparison results of IGD, C-metric, 1/HV and 1/PSP. In the table, the second, third, fourth and fifth columns represent the average values of 30 times of IGD, C-metric, 1/HV and 1/PSP respectively. In addition, Wilcoxon signed rank test at the 5% significance level was used, and the test results are given in the last row of each table. 'B/S/W' indicates that the effect of the proposed algorithm is significantly better than/basically similar to/significantly worse than the current algorithm.

From Tables 6 and 7, it can be seen that in terms of IGD, OL-DEVND is significantly better than INSGA_II in all 45 instances and better than HDMMODE in 32 instances. In terms of c-metric, OL-DEVND is significantly better than INSGA_II in 30 instances and better than HDMMODE in 28 instances. In terms of 1/HV and 1/PSP, OL-DEVND also shows good results. The main reason is that OL-DEVND is a global search algorithm with the highest probability of obtaining a highly convergent solution.

It can be obviously seen from Tables 6 and 7 that the difficulty of the problem increases with the increase of the number of customers and the decrease of the vehicle capacity. The reason is that the problem of having more customers and smaller capacity vehicles will have more path planning solutions, so it will be more difficult to converge on all goals. Moreover, any solution with a minimum value in any target is a non dominated solution, independent of the values of other targets. In the real scene, the preference for one goal may be higher than other goals. In the RVRP with four targets proposed in this paper, from a certain point of view, target f_2 (total driving distance) may be more important than other targets, because the driving distance is proportional to fuel consumption, so it has a direct impact on environmental pollution. In addition, for logistics companies, the target f_3(total travel time) and f_4(total cost) are equally important, because it is necessary to ensure that customers are served within the specified time to improve customer satisfaction, and to ensure that the total distribution cost is reduced throughout the logistics distribution process. Therefore, it is very important for RVRP to find the optimal value among all objectives by one method, because in most cases, the preference of decision makers is unknown a priori. This is analyzed in Table 8.

There are 45 instances in the data set, and the three algorithms are executed on each instance three times respectively. For each instance, a total of 30 groups of non dominated solutions can be obtained, and each algorithm generates a total of 1350 groups of non dominated solutions. In Table 3, each row except the last two rows represents a summary comparison of 30 runs of an instance. In terms of the number of runs, the minimum value of each target obtained by OL-DEVND is better (<), equal (=) or worse (>) than the minimum value of the corresponding target obtained by HDMMODE/INSGA_II. In the table, the first column names the instance name in the form of of"num_1- num_2- num_3", and and the second, third, fourth and fifth columns respectively list the comparison of the values of four objective functions in 30 runs. For example, the second row (instance "50-

Instance ID	HDMMODE		INSGA_II		ACO		ADPRA		Sim-BRIG-LS		OL-DEVND	
	f1	f2	f1	f2	f1	f2	f1	f2	f1	f2	f1	f2
P-n16-k8	13	536	11	517	12	521	10	503	11	502	9	487
P-n22-k8	14	723	12	691	13	703	12	692	11	673	9	654
P-n40-k5	9	521	10	532	9	517	8	496	7	489	6	477
P-n55-k8	13	652	12	639	12	643	11	626	10	612	9	597
P-n76-k4	8	642	7	636	7	639	6	631	6	625	5	611

Table 2. Comparison of different algorithm.

Instance ID	f1						f2					
	HDMMODE	INSGA_II	ACO	ADPRA	Sim-BRIG-LS	OL-DEVND	HDMMODE	INSGA_II	ACO	ADPRA	Sim-BRIG-LS	OL-DEVND
P-n16-k8	−1.15	−1.12	−1.10	−1.09	−1.02	-	−2.21	−2.20	−2.15	−2.13	−2.04	-
P-n22-k8	−1.12	−1.10	−1.06	−1.05	−1.03	-	−2.22	−2.21	−2.16	−2.10	−2.06	-
P-n40-k5	−1.11	−1.07	−1.05	−1.06	−1.03	-	−2.20	−2.19	−2.14	−2.14	−2.05	-
P-n55-k8	−1.10	−1.07	−1.08	−1.05	−1.02	-	−2.23	−2.20	−2.16	−2.09	−2.06	-
P-n76-k4	−1.08	−1.06	−1.05	−1.03	−1.01	-	−2.25	−2.19	−2.14	−2.11	−2.05	-

Table 3. T-test results.

Instance ID	HDMMODE				INSGA_II				ACO				ADPRA				Sim-BRIG-LS				OL-DEVND			
	f1	f2	f3	f4	f1	f2	f3	f4	f1	f2	f3	f4	f1	f2	f3	f4	f1	f2	f3	f4	f1	f2	f3	f4
50–0–2	18	703	1625	2335	17	689	1530	2103	15	658	1301	1925	13	620	1120	1789	11	596	1025	1526	10	587	985	1358
50–1–4	17	712	1635	2352	16	695	1535	2113	16	662	1309	1937	14	621	1123	1796	12	595	1035	1537	11	589	992	1365
50–2–3	19	709	1645	2368	18	697	1548	2126	16	649	1311	1952	14	619	1125	1810	11	598	1032	1540	10	592	996	1372
150–0–1	50	889	2102	2802	49	875	2024	2531	48	845	1825	2235	46	824	1625	2103	46	792	1525	1968	45	782	1203	1869
150–1–3	52	896	2109	2812	51	882	2043	2542	50	856	1842	2242	48	829	1634	2112	47	803	1527	1978	46	786	1208	1965
150–2–0	51	895	2117	2827	50	892	2052	2554	49	864	1853	2251	46	826	1642	2125	45	801	1531	1982	44	792	1210	1972
250–0–0	85	1502	2502	3869	83	1451	2301	3521	82	1345	2103	3201	82	1207	1965	3012	80	1102	1768	2865	78	1025	1550	2603
250–1–3	86	1510	2510	3878	82	1459	2310	3541	81	1355	2107	3213	80	1210	1968	3018	78	1105	1776	2883	76	1035	1549	2614
250–2–2	84	1508	2523	3882	82	1472	2325	3552	81	1368	2112	3225	80	1212	197	3025	76	1098	1781	2893	75	1029	1562	2635

Table 4. Results of different algorithms on real-world logistics dataset.

Instance ID	HDMMODE				INSGA_II				ACO				ADPRA				Sim-BRIG-LS			
	f1	f2	f3	f4	f1	f2	f3	f4	f1	f2	f3	f4	f1	f2	f3	f4	f1	f2	f3	f4
50–0–2	−1.12	−2.46	−5.03	−5.14	−1.09	−2.45	−5.02	−5.14	−1.08	−2.41	−5.02	−5.13	−1.05	−2.36	−5.02	−5.13	−1.02	−2.35	−5.01	−5.12
50–1–4	−1.10	−2.45	−5.03	−5.13	−1.07	−2.44	−5.03	−5.15	−1.07	−2.43	−5.03	−5.14	−1.06	−2.38	−5.02	−5.14	−1.05	−2.34	−5.02	−5.11
50–2–3	−1.11	−2.43	−5.04	−5.15	−1.06	−2.42	−5.02	−5.17	−1.05	−2.39	−5.01	−5.16	−1.04	−2.37	−5.03	−5.14	−1.08	−2.36	−5.01	−5.12
150–0–1	−1.09	−2.43	−5.06	−5.16	−1.10	−2.42	−5.02	−5.16	−1.10	−2.41	−5.02	−5.16	−1.06	−2.43	−5.04	−5.16	−1.12	−2.42	−5.02	−5.13
150–1–3	−1.11	−2.44	−5.03	−5.17	−1.11	−2.46	−5.03	−5.14	−1.12	−2.45	−5.03	−5.17	−1.08	−2.45	−5.04	−5.17	−1.15	−2.45	−5.03	−5.18
150–2–0	−1.13	−2.45	−5.04	−5.17	−1.12	−2.45	−5.01	−5.18	−1.06	−2.44	−5.01	−5.14	−1.10	−2.46	−5.03	−5.15	−1.13	−2.39	−5.04	−5.17
250–0–0	−1.12	−2.44	−5.06	−5.16	−1.19	−2.42	−5.02	−5.17	−1.12	−2.46	−5.04	−5.15	−1.11	−2.45	−5.02	−5.16	−1.16	−2.41	−5.02	−5.15
250–1–3	−1.13	−2.41	−5.04	−5.14	−1.13	−2.45	−5.04	−5.16	−1.13	−2.42	−5.05	−5.17	−1.09	−2.43	−5.01	−5.13	−1.08	−2.40	−5.01	−5.14
250–2–2	−1.12	−2.42	−5.03	−5.16	−1.12	−2.44	−5.03	−5.15	−1.14	−2.43	−5.02	−5.14	−1.12	−2.42	−5.03	−5.17	−1.09	−2.43	−5.03	−5.16

Table 5. T-test results.

0–1") and the third column (target) indicate that compared with HDMMODE, OL-DEVND can find the target value 18 times better, 0 times equal and 12 times worse in 30 runs. Compared with INSGA_II, OL-DEVND can find the target value 20 times better, 0 times equal and 10 times worse in 30 runs. The penultimate row provides the distribution of 1350 comparisons in the good, equal, and poor categories. The results show that for targets and, the minimum target value obtained by OL-DEVND is better than that of HDMMODE and INSGA_II, which is equal in the target f_2, but slightly worse in the target. The last line obtains the minimum value of each target in 30 runs of a method, and provides the comparison result statistics according to the number of instances between the three methods of each target. For the target, compared with HDMMODE, OL-DEVND can find a better value on 33 instances, an equal value on 2 instances, and a worse value on 10 instances. Compared with INSGA_II, OL-DEVND can find a better value on 37 instances, an equal value on 3 instances, and a worse value on 5 instances. Therefore, the last two rows of Table 3 clearly show that the performance of OL-DEVND is better than HDMMODE and INSGA_II in the single operation and the best operation of all 30 operations.

In order to further verify the efficiency of the algorithm proposed in this paper, we compared the algorithm proposed in this paper with ACO[40], ADPRA[41], Adaptive GA[43] and MODEA[44] on "50-0–3", "50-1–2", "150-0–1", "150-2–4", "250-1–1" and "250-2–3" (which are randomly selected). Similarly, the parameter settings of all comparison algorithms are consistent with the original references. All algorithms are carried out under equal conditions. Equal conditions mean using the same starting and termination criterion, equal number of starting search points, the same data set, the same hardware running the algorithms. The comparison results among different algorithms are shown in Table 9.

Instance	IGD		C-metric		1/HV		1/PSP	
	INSGA-II	OL-DEVND	C(IN, OL)	C(OL, IN)	INSGA-II	OL-DEVND	INSGA-II	OL-DEVND
50–0–0	0.001365	0.001037	0.488127	0.177221	1.252565	1.246721	0.052021	0.049303
50–0–1	0.004428	0.003819	0.046217	0.388213	2.304726	2.244080	0.051837	0.030474
50–0–2	0.005552	0.004603	0.059822	0.401222	2.680807	3.547332	0.050526	0.023793
50–0–3	0.002681	0.001817	0.492138	0.048821	1.617982	1.610031	0.050017	0.028712
50–0–4	0.005661	0.004432	0.331418	0.329812	1.064302	1.064320	0.048932	0.027096
50–1–0	0.001708	0.001003	0.600212	0.288053	1.219747	1.173467	0.094251	0.046426
50–1–1	0.004309	0.003218	0.047291	0.442018	2.303739	2.216229	0.084021	0.048201
50–1–2	0.005531	0.004217	0.051218	0.399471	2.732106	2.499200	0.091044	0.047902
50–1–3	0.002627	0.001507	0.552134	0.050133	1.610114	1.528755	0.088255	0.048201
50–1–4	0.005521	0.004318	0.434218	0.440781	1.082231	1.080949	0.086279	0.047709
50–2–0	0.007097	0.005667	0.776521	0.287541	1.113149	1.109540	0.098214	0.052876
50–2–1	0.004804	0.003619	0.048210	0.461920	1.408377	1.361914	0.097317	0.052909
50–2–2	0.004783	0.003657	0.288072	0.284718	1.623250	1.126961	0.098012	0.053883
50–2–3	0.004008	0.002892	0.442026	0.053346	1.371167	1.423129	0.097823	0.054072
50–2–4	0.009208	0.007015	0.809217	0.005825	1.145454	1.120387	0.097204	0.052102
150–0–0	0.001769	0.001058	0.562016	0.387209	1.324324	1.287800	0.062375	0.053128
150–0–1	0.006407	0.005318	0.004088	0.552017	3.335379	3.620381	0.063561	0.053208
150–0–2	0.007098	0.005735	0.005821	0.621720	3.905777	3.340716	0.064426	0.054082
150–0–3	0.002817	0.001517	0.202671	0.362819	2.815973	1.810984	0.063890	0.055231
150–0–4	0.008817	0.007628	0.020122	0.802088	2.512430	2.468362	0.065286	0.054027
150–1–0	0.001816	0.001084	0.382117	0.299024	1.322730	1.255936	0.077423	0.055324
150–1–1	0.006218	0.005042	0.003629	0.552014	3.411048	3.468417	0.078211	0.056438
150–1–2	0.007163	0.005859	0.005528	0.682007	3.915841	3.375744	0.079246	0.057903
150–1–3	0.002887	0.001628	0.225312	0.248231	2.815973	1.810984	0.078305	0.058023
150–1–4	0.008817	0.007626	0.021025	0.799256	2.509933	2.370814	0.079601	0.060214
150–2–0	0.002372	0.001106	0.499207	0.242019	1.217176	1.184100	0.086426	0.065241
150–2–1	0.006738	0.005648	0.000921	0.642011	2.671568	2.498857	0.085231	0.065292
150–2–2	0.007412	0.006219	0.002028	0.556281	4.055347	3.861093	0.087262	0.066620
150–2–3	0.002866	0.001537	0.098231	0.244833	2.763790	1.806917	0.087710	0.066391
150–2–4	0.008874	0.007537	0.011528	0.834120	2.025390	1.848545	0.088238	0.067202
250–0–0	0.002669	0.001362	0.253081	0.500726	1.360541	1.320985	0.102654	0.071343
250–0–1	0.006738	0.005317	0.002520	0.634621	4.081666	3.605527	0.110435	0.072054
250–0–2	0.007664	0.006628	0.002239	0.530925	4.582174	3.746623	0.112467	0.073025
250–0–3	0.005021	0.003912	0.162073	0.162242	1.775480	2.696668	0.114371	0.073531
250–0–4	0.010538	0.008727	0.010216	0.552017	2.118967	2.576483	0.115069	0.074092
250–1–0	0.002678	0.001218	0.237615	0.490216	1.360594	1.329050	0.140832	0.089215
250–1–1	0.007738	0.006421	0.002107	0.620971	4.087104	3.469861	0.144231	0.090207
250–1–2	0.007698	0.006206	0.002103	0.612714	4.582174	3.932410	0.145078	0.090620
250–1–3	0.005027	0.003719	0.090142	0.156820	1.775480	1.717567	0.143321	0.091072
250–1–4	0.010509	0.008147	0.009821	0.552781	2.119452	2.025460	0.145109	0.092082
250–2–0	0.002318	0.001128	0.260736	0.334521	1.276380	1.246856	0.134027	0.091453
250–2–1	0.007065	0.005735	0.001009	0.482351	4.312575	3.845252	0.142518	0.090112
250–2–2	0.007429	0.006107	0.001025	0.392717	4.804782	3.608779	0.136068	0.092013
250–2–3	0.005098	0.003917	0.099758	0.152094	1.774576	1.699955	0.141072	0.092107
250–2–4	0.012289	0.009898	0.096087	0.112864	2.329911	2.169099	0.138090	0.092208
B/S/W	45/0/0		30/5/10		20/5/20		45/0/0	

Table 6. Average values of IGD, C-metric, 1/HV, and 1/PSP of OL-DEVND, INSGA_II.

As can be seen from Table 4, In terms of IGD, 1/HV, and 1/PSP, OL-DEVND is superior to other algorithms in the selected random instances, and in terms of C-metric, OL-DEVND also shows a good effect.

If the non dominated solution set found by the former algorithm is better than the latter algorithm in convergence and diversity, it is considered that the former algorithm is better than the other one. Because visual representation is easy to understand, it is usually used to compare the convergence and diversity of solution sets obtained by different methods. In order to intuitively show the convergence and diversity of non dominated solutions obtained by OL-DEVND, HDMMODE and INSGA_II, we use heat map visualization for analysis. The

Instance	IGD		C-metric		1/HV		1/PSP	
	HDMMODE	OL-DEVND	C(MM, OL)	C(OL, MM)	HDMMODE	OL-DEVND	HDMMODE	OL-DEVND
50–0–0	0.001245	0.001037	0.501247	0.154721	1.249505	1.246721	0.050034	0.049303
50–0–1	0.003961	0.003819	0.052172	0.340223	2.430157	2.244080	0.051025	0.030474
50–0–2	0.004619	0.004603	0.062177	0.356625	2.550903	3.547332	0.048017	0.023793
50–0–3	0.002032	0.001817	0.533371	0.042178	1.614943	1.610031	0.046092	0.028712
50–0–4	0.005007	0.004432	0.365316	0.306011	1.049835	1.064320	0.042182	0.027096
50–1–0	0.001121	0.001003	0.641728	0.241323	1.214223	1.173467	0.084201	0.046426
50–1–1	0.003671	0.003218	0.052410	0.400217	2.225878	2.216229	0.079837	0.048201
50–1–2	0.005204	0.004217	0.057812	0.360612	2.589379	2.499200	0.088252	0.047902
50–1–3	0.002137	0.001507	0.581248	0.042133	1.585007	1.528755	0.084218	0.048201
50–1–4	0.005105	0.004318	0.445719	0.43671	1.076518	1.080949	0.081025	0.047709
50–2–0	0.006013	0.005667	0.831821	0.223431	1.095419	1.109540	0.089142	0.052876
50–2–1	0.004217	0.003619	0.055431	0.416662	1.384432	1.361914	0.090561	0.052909
50–2–2	0.003821	0.003657	0.301022	0.289001	1.567838	1.126961	0.089362	0.053883
50–2–3	0.002993	0.002892	0.501266	0.049937	1.350953	1.423129	0.091936	0.054072
50–2–4	0.007821	0.007015	0.912018	0.004638	1.136083	1.120387	0.090102	0.052102
150–0–0	0.001092	0.001058	0.621121	0.331752	1.318510	1.287800	0.058841	0.053128
150–0–1	0.005202	0.005318	0.005527	0.521218	3.331812	3.620381	0.059621	0.053208
150–0–2	0.005829	0.005735	0.006431	0.582109	3.783322	3.340716	0.060172	0.054082
150–0–3	0.001993	0.001517	0.244977	0.328444	2.744297	1.810984	0.059206	0.055231
150–0–4	0.008026	0.007628	0.023102	0.764000	2.492019	2.468362	0.058934	0.054027
150–1–0	0.001341	0.001084	0.420428	0.260921	1.297301	1.255936	0.070128	0.055324
150–1–1	0.005468	0.005042	0.004421	0.520002	3.196890	3.468417	0.069105	0.056438
150–1–2	0.006219	0.005859	0.005683	0.663003	3.749770	3.375744	0.070013	0.057903
150–1–3	0.002102	0.001628	0.232892	0.243006	2.776405	1.810984	0.070128	0.058023
150–1–4	0.007908	0.007626	0.023407	0.772341	2.492044	2.370814	0.071521	0.060214
150–2–0	0.001762	0.001106	0.588713	0.181712	1.203338	1.184100	0.079065	0.065241
150–2–1	0.005819	0.005648	0.000992	0.601027	2.614605	2.498857	0.078563	0.065292
150–2–2	0.006453	0.006219	0.002319	0.523366	4.015726	3.861093	0.079632	0.066620
150–2–3	0.002012	0.001537	0.099129	0.223109	2.595272	1.806917	0.081539	0.066391
150–2–4	0.008013	0.007537	0.012417	0.790021	1.998753	1.848545	0.080016	0.067202
250–0–0	0.002073	0.001362	0.261221	0.477129	1.353359	1.320985	0.098352	0.071343
250–0–1	0.006012	0.005317	0.002870	0.598021	3.949198	3.605527	0.106326	0.072054
250–0–2	0.006934	0.006628	0.002618	0.499021	4.269545	3.746623	0.107321	0.073025
250–0–3	0.004452	0.003912	0.120721	0.142017	1.711368	2.696668	0.109362	0.073531
250–0–4	0.009458	0.008727	0.012638	0.507200	2.078600	2.576483	0.108902	0.074092
250–1–0	0.002238	0.001218	0.260012	0.450127	1.353134	1.329050	0.125427	0.089215
250–1–1	0.006932	0.006421	0.002366	0.590128	3.998209	3.469861	0.128891	0.090207
250–1–2	0.006892	0.006206	0.002409	0.582103	4.517589	3.932410	0.131098	0.090620
250–1–3	0.004109	0.003719	0.098231	0.134002	1.762584	1.717567	0.132012	0.091072
250–1–4	0.009829	0.008147	0.010921	0.505211	2.044450	2.025460	0.132088	0.092082
250–2–0	0.001834	0.001128	0.301288	0.288127	1.265619	1.246856	0.128902	0.091453
250–2–1	0.005892	0.005735	0.001213	0.452129	4.147949	3.845252	0.130173	0.090112
250–2–2	0.006671	0.006107	0.001342	0.362819	4.701922	3.608779	0.129983	0.092013
250–2–3	0.004246	0.003917	0.112083	0.123652	1.764506	1.699955	0.130128	0.092107
250–2–4	0.010012	0.009898	0.010281	0.510286	2.318808	2.169099	0.129983	0.092208
B/S/W	32/13/0		28/7/10		16/23/6		44/1/0	

Table 7. Average values of IGD, HV, 1/HV, and 1/PSP of OL-DEVND, HDMMODE.

heat map provides a visual representation of the solution set for many target problems and helps to observe the trade-offs between various targets in a clear way. The heat map displays the data as a grid of pixels whose color represents the proportional value from maximum (hot) to minimum (cold)[45]. In the heat map representation, each row represents a solution and each column represents a goal. The color of the cell represents the target value for a particular solution. The cold color indicates the convergence of the solution, and the distribution of various colors indicates the diversity of the solution. If the heat map of the solution shows the colder color and the distribution of the whole color range, the non dominated solution obtained by one method can be regarded

Instance	f1			f2			f3			f4		
	<	=	>	<	=	>	<	=	>	<	=	>
50–0–0	0/0	30/30	0/0	10/12	10/11	10/7	0/0	30/30	0/0	10/12	10/11	10/7
50–0–1	22/24	6/6	2/0	18/20	0/0	12/10	0/0	30/30	0/0	17/19	0/0	13/11
50–0–2	0/0	30/30	0/0	17/19	0/0	13/11	0/0	30/30	0/0	18/20	0/0	12/10
50–0–3	0/0	30/30	0/0	11/14	0/0	19/16	0/0	30/30	0/0	10/12	0/0	20/18
50–0–4	0/0	30/30	0/0	17/18	1/3	12/9	0/0	30/30	0/0	16/19	1/3	13/8
50–1–0	0/0	30/30	0/0	20/21	0/0	10/9	0/0	30/30	0/0	22/23	0/0	8/7
50–1–1	16/18	12/11	2/1	17/19	2/4	11/7	0/0	30/30	0/0	18/19	2/3	10/8
50–1–2	0/0	30/30	0/0	11/13	0/0	19/17	0/0	30/30	0/0	12/13	0/0	18/17
50–1–3	0/0	30/30	0/0	11/13	0/0	19/17	0/0	30/30	0/0	11/13	0/0	19/17
50–1–4	0/0	30/30	0/0	20/23	1/3	9/4	0/0	30/30	0/0	21/23	1/3	8/4
50–2–0	0/0	30/30	0/0	25/27	0/0	5/3	0/0	30/30	0/0	24/27	0/0	4/3
50–2–1	0/0	30/30	0/0	25/27	0/0	5/3	0/0	30/30	0/0	25/26	0/0	5/4
50–2–2	0/0	30/30	0/0	27/29	0/0	3/1	0/0	30/30	0/0	27/29	0/0	3/1
50–2–3	0/0	30/30	0/0	24/25	0/0	6/5	0/0	30/30	0/0	25/26	0/0	5/4
50–2–4	0/0	30/30	0/0	30/30	0/0	0/0	0/0	30/30	0/0	30/30	0/0	0/0
150–0–0	10/14	16/16	4/0	24/25	0/0	6/5	16/18	11/12	3/0	23/26	0/0	7/4
150–0–1	0/0	30/30	0/0	23/25	0/0	7/5	30/30	0/0	0/0	23/24	0/0	7/6
150–0–2	0/0	30/30	0/0	28/30	0/0	2/0	30/30	0/0	0/0	27/30	0/0	3/0
150–0–3	0/0	30/30	0/0	28/30	0/0	2/0	0/0	30/30	0/0	28/30	0/0	2/0
150–0–4	0/0	6/12	24/18	22/26	0/0	8/4	7/9	23/21	0/0	22/25	0/0	8/5
150–1–0	8/11	13/16	9/3	24/25	0/0	6/5	16/18	11/12	3/0	24/25	0/0	6/5
150–1–1	0/0	30/30	0/0	23/25	0/0	7/5	30/30	0/0	0/0	23/26	0/0	7/4
150–1–2	0/0	30/30	0/0	28/30	0/0	2/0	30/30	0/0	0/0	27/30	0/0	3/0
150–1–3	0/0	30/30	0/0	29/30	0/0	1/0	0/0	30/30	0/0	29/30	0/0	1/0
150–1–4	0/0	6/8	24/22	22/24	0/0	8/6	7/9	23/21	0/0	21/24	0/0	9/6
150–2–0	0/0	30/30	0/0	16/18	0/0	14/12	20/22	10/8	0/0	15/17	0/0	15/13
150–2–1	0/0	30/30	0/0	24/25	0/0	6/5	30/30	0/0	0/0	22/25	0/0	8/5
150–2–2	0/0	30/30	0/0	28/30	0/0	2/0	30/30	0/0	0/0	28/29	0/0	2/1
150–2–3	0/0	30/30	0/0	28/30	0/0	2/0	0/0	30/30	0/0	28/30	0/0	2/0
150–2–4	0/0	30/30	0/0	24/25	0/0	6/5	4/7	26/23	0/0	21/25	0/0	9/5
250–0–0	0/0	30/30	0/0	9/12	0/0	21/18	22/24	5/5	3/1	9/11	0/0	21/19
250–0–1	3/7	18/19	9/4	22/23	0/0	8/7	30/30	0/0	0/0	20/23	0/0	10/7
250–0–2	0/0	30/30	0/0	29/30	0/0	1/0	30/30	0/0	0/0	27/30	0/0	3/0
250–0–3	0/0	30/30	0/0	30/30	0/0	0/0	0/0	30/30	0/0	30/30	0/0	0/0
250–0–4	0/0	4/6	26/24	25/27	0/0	5/3	25/26	5/4	0/0	23/27	0/0	7/3
250–1–0	0/0	30/30	0/0	9/12	0/0	21/18	22/24	5/5	3/1	9/13	0/0	21/17
250–1–1	4/7	20/21	6/2	22/23	0/0	8/7	30/30	0/0	0/0	19/23	0/0	11/7
250–1–2	0/0	30/30	0/0	29/30	0/0	1/0	30/30	0/0	0/0	26/30	0/0	4/0
250–1–3	0/0	30/30	0/0	30/30	0/0	0/0	0/0	30/30	0/0	30/30	0/0	0/0
250–1–4	0/0	4/7	26/23	25/27	0/0	5/3	25/26	5/4	0/0	23/27	0/0	7/3
250–2–0	0/0	30/30	0/0	18/20	0/0	12/10	24/27	3/2	3/1	18/22	0/0	12/8
250–2–1	2/5	22/24	6/1	24/25	0/0	6/5	30/30	0/0	0/0	22/25	0/0	8/5
250–2–2	1/5	27/23	2/2	30/30	0/0	0/0	30/30	0/0	0/0	30/30	0/0	0/0
250–2–3	0/0	30/30	0/0	30/30	0/0	0/0	0/0	30/30	0/0	30/30	0/0	0/0
250–2–4	0/0	4/6	26/23	17/19	0/0	13/11	0/0	30/30	0/0	16/19	0/0	14/11
Total	66/91	1126/1137	158/122	1003/1097	14/21	333/232	548/570	765/778	15/2	928/1077	14/20	408/253
Best	1/2	44/43	0/0	34/37	2/4	9/4	14/17	31/28	0/0	33/37	2/3	10/5

Table 8. Comparison of OL-DEVND with HDMMODE/INSGA_II in terms of count of the runs as well as in overall 30 runs, on which OL-DEVND achieved better (<), equal (=) and worse (>) values in four objectives.

as a good solution. In order to better use the heat map for visual representation, all targets must have the same scale. Therefore, we standardized all target values to the range of [0,1]. Each heat map is a visual representation of the non dominated solution obtained by the method of a specific instance in all 30 runs. Each target value is displayed in a specific color in the cold (blue) to hot (red) range.

Indexes	Algorithms	50-0–3	50-1–2	150-0–1	150-2–4	250-1–1	250-2–3
IGD	ACO	0.004037	0.006822	0.007521	0.008533	0.008342	0.005902
	ADPRA	0.004861	0.007069	0.007962	0.009347	0.008828	0.006635
	Adaptive GA	0.003463	0.006347	0.007348	0.008219	0.007619	0.005617
	MODEA	0.002039	0.005138	0.006036	0.008342	0.006831	0.004528
	OL-DEVND	0.001817	0.004217	0.005318	0.007537	0.006421	0.003917
C-metric	C(EI, AC)	0.632081	0.120872	0.366521	0.083071	0.062761	0.361931
	C(AC, EI)	0.562172	0.108216	0.293829	0.082863	0.054212	0.275324
	C(EI, AD)	0.092017	0.291873	0.093702	0.192762	0.565291	0.092774
	C(AD, EI)	0.065291	0.233082	0.073936	0.011724	0.486221	0.065372
	C(EI, GA)	0.454296	0.652692	0.386542	0.558862	0.783553	0.010286
	C(GA, EI)	0.392751	0.397697	0.297394	0.398625	0.645361	0.010293
	C(EI, MO)	0.342607	0.058621	0.201296	0.135820	0.562072	0.235612
	C(MO, EI)	0.321024	0.046102	0.017524	0.098601	0.012764	0.152976
1/HV	ACO	3.096192	4.538216	5.631983	3.521093	5.210732	3.885201
	ADPRA	3.371937	5.001741	5.996521	3.891025	5.702810	4.221094
	Adaptive GA	2.887915	3.965286	5.137639	3.120073	4.802655	3.470638
	MODEA	2.014364	3.227315	4.541731	2.523108	4.092071	2.580925
	OL-DEVND	1.610031	2.499200	3.620381	1.848545	3.469861	1.699955
1/PSP	ACO	0.042093	0.066525	0.067308	0.081044	0.112034	0.110537
	ADPRA	0.045075	0.070271	0.072034	0.084092	0.120846	0.121026
	Adaptive GA	0.038742	0.061968	0.063581	0.078360	0.100163	0.102074
	MODEA	0.033891	0.053096	0.059603	0.072083	0.098325	0.098673
	OL-DEVND	0.028712	0.047902	0.053208	0.067202	0.090207	0.092107

Table 9. The comparison results among different algorithms. Note: C(OL, AC): C(OL-DEVND, ACO) C(AC, OL): C(ACO, OL-DEVND) C(OL, AD): C(OL-DEVND, ADPRA). C(AD, OL): C(ADPRA, OL-DEVND) C(OL, GA): C(OL-DEVND, Adaptive GA) C(GA, OL): C(Adaptive GA, OL-DEVND). C(OL, MO): C(OL-DEVND, MODEA) C(MO, OL): C(MODEA, OL-DEVND).

In the test data set, the more customers and the smaller vehicle capacity, the more difficult it is to deal with the problem. Therefore, we selected the data examples of $num\ 1 = 250$ (i.e. 250 customers) and $num\ 2 = 2$ (i.e. the third category of vehicles with the smallest capacity compared with other categories) to visually compare the non dominant solutions obtained by the three algorithms.

Figure 5 shows the convergence and diversity of the non dominated solutions obtained by the three algorithms using the heat map in five cases, namely: 250-2–0, 250-2–1, 250-2–2, 250-2–3 and 250-2–4. The first, second and third lines of the heat map represent the non dominated solutions obtained by OL-DEVND, HDMMODE and INSGA_II on the selected data instance respectively. The rows of each heat map have been rearranged in ascending order of f_2. In all five instances, the heat map of OL-DEVND shows significantly colder colors than the HDMMODE and INSGA_II heat maps. Therefore, in all five instances OL-DEVND has better convergence in f_2 than HDMMODE and INSGA_II.

From all the comparison results, it can be seen that HDMMODE and INSGA_II always lack different solutions, which is more obvious in goals f_1 and f_2. In addition, the heat map of OL-DEVND has better convergence than HDMMODE and INSGA_II in terms of targets f_2 and f_3. Obviously, the goals conflict with each other, so a method is better if it produces a better diversity of non dominated solutions, that is, if the non dominated solutions have a wider distribution over the entire color range of all goals. Therefore, heat map visualization shows that the performance of OL-DEVND is better than that of HDMMODE and INSGA_II in terms of convergence and diversity.

In order to show the heat map more clearly, that is, the convergence and diversity of the obtained non dominated solutions. Taking "250-2–4" as an example, Fig. 6 shows the heat map normalization curves of 30 experiments of OL-DEVND, HDMMODE and INSGA_II on f_2.

It can be seen from Fig. 6 that the normalized values of OL-DEVND in 30 experiments are all smaller than HDMMODE and INSGA_II. And the curve variation amplitude of OL-DEVND is larger than that of HDMMODE and INSGA_II. This indicates that OL-DEVND has good convergence and diversity. The experiment results of the optimal values show that the OL-DEVND has better global optimization performance.

Conclusion

In order to make VRP more realistic, this paper considers RVRP under four constraints: complex road network constraints, capacity constraints, time window constraints, and demand separability constraints. In this problem, in order to meet the planning needs of different decision makers for vehicle paths under different constraints, multiple equivalent optimal paths are sought. RVRP is regarded as a MMOP, and a method combining DE and VND is designed. Firstly, in order to expand the search range of the population, OL is introduced into the basic

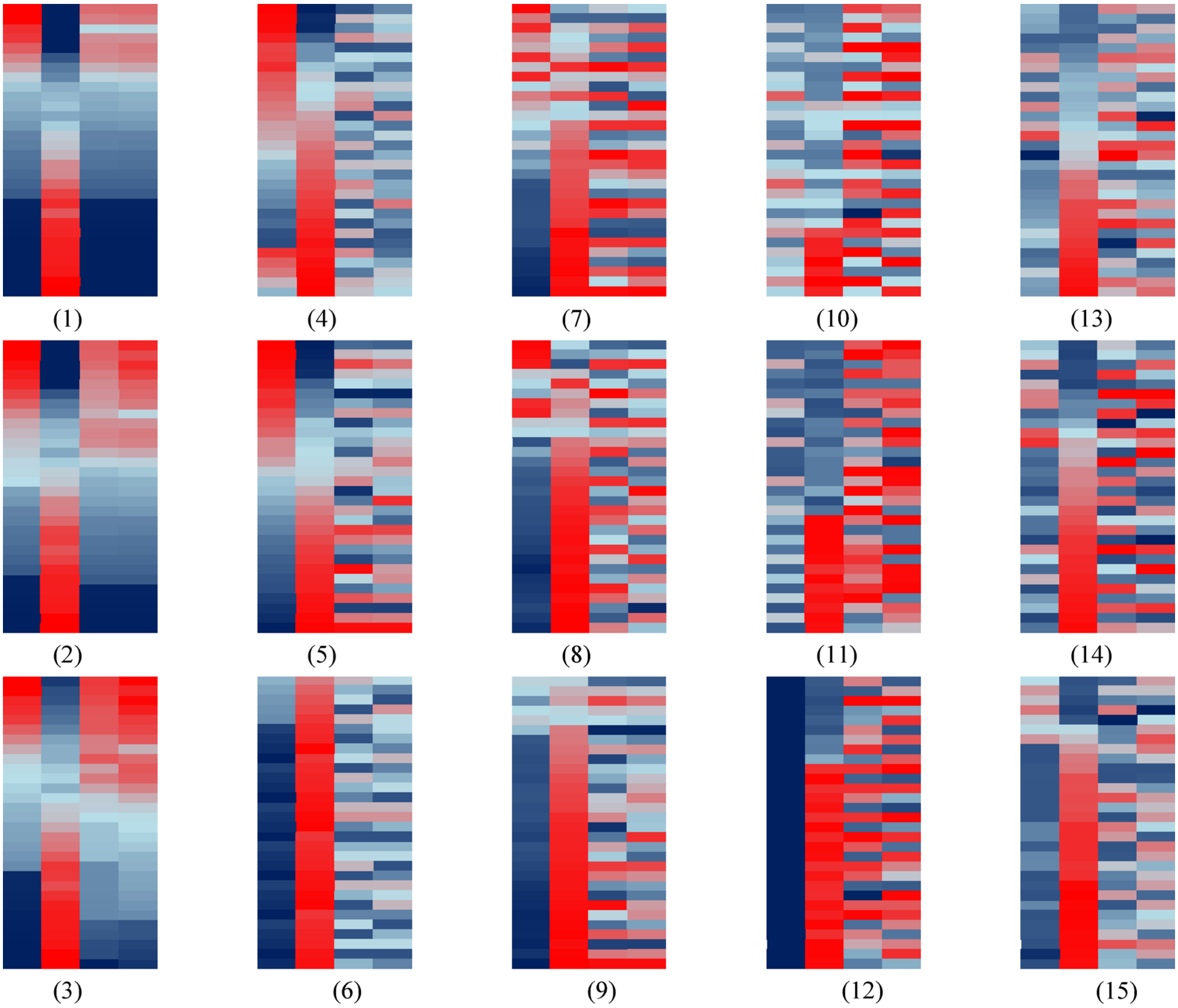

Fig. 5. Heat-maps of non-dominated solutions obtained by OL-DEVND, HDMMODE and INSGA_II on selected instances: (1) OL-DEVND on 250-2–0, (2) HDMMODE on 250-2–0, (3) INSGA_II on 250-2–0, (4) OL-DEVND on 250-2–1, (5) HDMMODE on 250-2–1, (6) INSGA_II on 250-2–1, (7) OL-DEVND on 250-2–2, (8) HDMMODE on 250-2–2, (9) INSGA_II on 250-2–2, (10) OL-DEVND on 250-2–3, (11) HDMMODE on 250-2–3, (12) INSGA_II on 250-2–3, (13) OL-DEVND on 250-2–4, (14) HDMMODE on 250-2–4, (15) INSGA_II on 250-2–4.

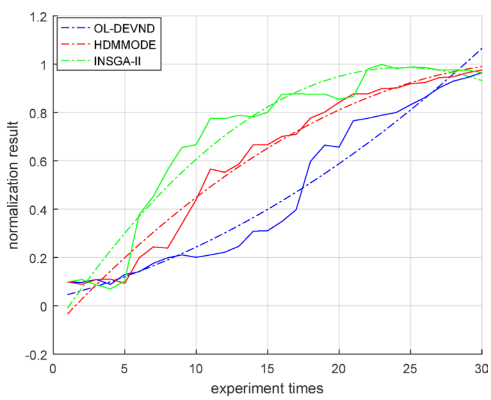

Fig. 6. Heat map normalization curves of different algorithms.

differential evolution algorithm to broaden the search range of the solution. Secondly, to address the issue of DE being prone to premature convergence and falling into local optima, a VND is embedded to enhance the population search capability. Experimental simulations were conducted on the RVRP data-set, and the results showed that the non dominated solution set obtained by this method was superior to HDMMODE and INSGA_II in terms of convergence and diversity. This method achieved the best overall performance, thus verifying the effectiveness of the proposed OL-DEVND compared to existing methods. The OL-DEVND proposed in this article can obtain multiple equivalent optimal paths when solving RVRP, providing reference inspiration for RVRP and its variants.

The OL-DEVND framework offers actionable solutions for real-world logistics and urban mobility challenges. For example, logistics firms (e.g., e-commerce, cold-chain) can integrate OL-DEVND into fleet management systems to dynamically adjust routes under fluctuating demands and traffic conditions. For customers with large orders, the algorithm's demand-splitting capability allows partial deliveries via multiple vehicles, maximizing load utilization. By optimizing routes under complex road network constraints, OL-DEVND can reduce peak-hour traffic density in urban corridors.

However, in subsequent research, it is still necessary to consider local searches that meet specific objectives and improve the performance of the algorithm. Due to the uncertainty of vehicle operation and distribution order, for example, temporary distribution will occur, which will affect the route planning. Therefore, the dynamic vehicle routing problem and multiple equivalent optimal the path planning problem will be further studied to further improve the robustness in the future.

Data availability

Data is provided within the manuscript : All data generated or analysed during this study are included in https://github.com/psxjpc/and http://vrp.galgos.inf.puc-rio.br/index.php/en/.

Received: 16 December 2024; Accepted: 11 June 2025
Published online: 25 September 2025

References

1. Danting, G. B. & Ramser, J. H. The truck dispatching problem[J]. *Manage. Sci.* **6** (1), 80–91 (1959).
2. Pellegrini, P., Favaretto, D. & Moretti, E. Multiple Ant Colony Optimization for a Rich Vehicle Routing Problem: A Case Study[C]. Knowledge-based Intelligent Information & Engineering Systems & the XVII Italian Workshop on Neural Networks on International Conference, 627–634. (2007).
3. Pisinger, D. & Ropke, S. A general heuristic for vehicle routing problems[J]. *Comput. Oper. Res.* **34** (8), 2403–2435 (2007).
4. Goel, A. & Gruhn, V. A general vehicle routing Problem[J]. *Eur. J. Oper. Res.* **191** (3), 650–660 (2008).
5. Subramanian, A., Uchoa, E. & Ochi, L. S. *A Hybrid Algorithm for a Class of Vehicle Routing problems[J]*402519–2531 (Computers & Operations Research, 2013). 10.
6. Caceres-Cruz, J. et al. Rich vehicle routing problem: survey. *ACM Comput. Surveys*. **47** (2), Article32 (2015).
7. Rahma & Lahyani Mahdi khemakhem,frédéric semet. Rich vehicle routing problems: from a taxonomy to a definition[J]. *Eur. J. Oper. Res.* **241** (1), 1–14 (2015).
8. Qi, M., Peng, C. & Huang, X. General Metaheuristic Algorithm for a Set of Rich Vehicle Routing Problems[J]. *Transp. Res. Record J. Transp. Res. Board.*, **2548**(2548):97–106. (2016).
9. Rabbouch, B., Sadaoui, F. & Rafaa, M. Efficient implementation of the genetic algorithm to solve rich vehicle routing problems[J]. *Oper. Res. Int. Journal*, **21**(2021), 1763–1791 (2021).
10. Liang, J., Lin, H. Y., Yue, C. T. & Suganthan, P. N. Wang. multiobjective differential evolution for Higher-Dimensional multimodal multiobjective Optimization[J]. *IEEE/CAA J. Automatica Sinica*. **11** (6), 1–18 (2024).
11. Liang, J., Lin, H. Y., Yue, C. T., Yu, K. J. & Guo, Y. Qiao. Multiobjective differential evolution with speciation for constrained multimodal multiobjective optimization[J]. *IEEE Trans. Evol. Comput.* **27** (4), 1115–1112 (2023).
12. Li*, G. Q. et al. Zhang. A ring-hierarchy-based evolutionary algorithm for multimodal multi-objective optimization[J]. *Swarm Evol. Comput.* **81**, 101352 (2023).
13. LI, W. et al. Weighted indicator-based evolutionary algorithm for multimodal multiobjective optimization[J]. *IEEE Trans. Evol. Comput.* **25** (6), 1064–1078 (2021).
14. WANG, W. et al. Clearing based multimodal multi-objective evolutionary optimization with layer-to-layer strategy[J]. *Swarm Evol. Comput.* **68**, 100976 (2022).
15. YAO, X. et al. Multimodal multi-objective evolutionary algorithm for multiple path planning[J]. *Comput. Ind. Eng.* **169**, 108145 (2022).
16. Ming, F. et al. Balancing convergence and diversity in objective and decision spaces for multimodal Multi-Objective Optimization[J. *IEEE Trans. Emerg. Top. Comput. Intell.* **7** (2), 474–486 (2023).
17. Li, W. et al. Hierarchy ranking method for multimodal multi-objective optimization with local Pareto fronts[J]. *IEEE Trans. Evol. Comput.* **27** (1), 98–110 (2023).
18. Qu, B. Y., Li, G. S., Yan, L., Liang, J. & Yue, C. T. Yu, O. D. Crisalle. A grid-guided particle swarm optimizer for multimodal multi-objective problems[J]. *Appl. Soft Comput.* **117**, 108381 (2022).
19. Li, X., Liu, J. & ,Xie, Y. G. P. A Multi-modal Attention Graph Network with Dynamic Routing-By-Agreement for multi-label emotion recognition[J]. Knowledge-based systems, 283 (Jan.11) :1.1–1.14. (2024).
20. Oakey, A., Martinez-Sykora, A. & Cherrett, T. Improving the efficiency of patient diagnostic specimen collection with the aid of a multi-modal routing algorithm[J].Computers & operations research, 157(Sep.):1.1–1.24. (2023).
21. Srivastava, G., Singh, A. & Mallipeddi, R. NSGA-II with objective-specific variation operators for multi-objective vehicle routing problem with time windows[J]. *Expert Syst. Appl.* **176**, 114779 (2021).
22. Konstantakopoulos, G. D. et al. Delivering and picking goods under time window restrictions: an effective evolutionary algorithm[J]. *J. Intell. Fuzzy Syst.* **40**(3), 5323–5336 (2021).
23. Sethanan, K. & Jamrus, T. Hybrid differential evolution algorithm and genetic operator for multi-trip vehicle routing problem with back-hauls and heterogeneous fleet in the beverage logistics industry. *Sci. Direct[J] Computers Industrial Eng.* **146**, 106571 (2020).
24. Peng, B. et al. Solving the Multi-Depot green vehicle routing problem by a hybrid evolutionary Algorithm[J]. *Sustainability* **12** (5), 2127 (2020).
25. Sathyamurthy, E., Herrmann, J. W. & Azarm, S. .Hybrid Metaheuristic Approaches for the Multi-Depot Rural Postman Problem With Rechargeable and Reusable *Vehicles[J] IEEE Access.*, **12**(000):18. DOI:https://doi.org/10.1109/ACCESS.2024.3414331. (2024).

26. Ferreira, K. M. et al. A variable neighborhood search for the green vehicle routing problem with two-dimensional loading constraints and split delivery[J]. *Eur. J. Oper. Res.* (2), 316. https://doi.org/10.1016/j.ejor.2024.01.049 (2024).
27. Wu, T. et al. A RankNet-Inspired Surrogate-Assisted hybrid metaheuristic for expensive coverage Optimization[J]. *Computer Science- Neural and Evolutionary Computing*. Feb.(v2). https://doi.org/10.48550/arXiv.2501.07375 (2025).
28. Narasimhan, G. & Victor, A. .A hybrid approach with metaheuristic optimization and random forest in improving heart disease prediction[J].Scientific Reports[2025-05-13].https://doi.org/10.1038/s41598-024-73867-x
29. Liu, H. et al. *Evolutionary Multitasking Memetic Algorithm for Distributed Hybrid Flow-Shop Scheduling Problem with Deterioration Effect [J]* (IEEE Transactions on Automation Science and Engineering, 2024).
30. Liu, H. et al. Multi-Constraint distributed terminal distribution path planning for fresh agricultural products [J]. *Appl. Intell.* **55**(2), 180:1–19. https://doi.org/10.1007/s10489-024-06076-8 (2025).
31. Bai, Q. et al. Vehicle routing Problem for cold chain logistics based on data fusion technology to predict travel time[J].*Oper. Res. Int. Journal*, **24**(4).DOI:https://doi.org/10.1007/s12351-024-00851-8. (2024).
32. Yue, C. et al. Differential evolution using improved crowding distance for Multi-modal Multi-objective Optimization[J]. *Swarm Evol. Comput.* **62** (9), 100849 (2021).
33. Yu, X. P., Hu, Y. S. & Wu, P. The consistent vehicle routing problem considering driver equity and flexible route consistency[J]. Computers & Industrial Engineering, 187(Jan.):109803.1-109803.13. (2024).
34. Venu, D. et al. An efficient low complexity compression based optimal homomorphic encryption for secure fiber optic communication[J]. *Optik* **252**, 168545–. https://doi.org/10.1016/j.ijleo.2021.168545 (2022).
35. Augerat, P. et al. *Computational Results with a Branch and Cut Code for the Capacitated Vehicle Routing problem[J]*495 (Rapport de recherche - IMAG, 1995).
36. Zhou, Y. & Wang, J. A local Search-Based Multi-objective optimization algorithm for Multi-objective vehicle routing problem with time Windows[J]. *IEEE Syst. J.* **9** (3), 1100–1113 (2017).
37. Zhang, Q. & Li, H. MOEA/D: A Multi-objective evolutionary algorithm based on Decomposition[J]. *IEEE Trans. Evol. Comput.* **11** (6), 712–731 (2008).
38. Zitzler, E. & Thiele, L. Multi-objective evolutionary algorithms: a comparative case study and the strength Pareto approach[J]. *IEEE Trans. Evol. Comput.* **3** (4), 257–271 (1999).
39. Patil, R. S. & Shingan, G. G. A Parallel Non-dominated Sorting Genetic Algorithm-II for Vehicle Routing Problem in Supply Chain[C]//International Conference on Computational Intelligence.Springer, Singapore, (2024). https://doi.org/10.1007/978-981-97-3526-6_25
40. Starzec, G. et al. Ant colony optimization using two-dimensional pheromone for single-objective transport problems[J].Journal of computational science, 79(Jul.):1.1–1.9. (2024).
41. Wu, Y. A. & Dynamic Stochastic cumulative capacitated vehicle routing Problem[J].Asia-Pacific. *J. Oper. Res.* **42** (2). https://doi.org/10.1142/S0217595924500088 (2025).
42. Abdullahi, H. et al. A reliability-extended simheuristic for the sustainable vehicle routing problem with stochastic travel times and demands[J]. *J. Heuristics.* **31** (2). https://doi.org/10.1007/s10732-025-09555-4 (2025).
43. Ouertani, N. et al. A multi-compartment VRP model for the health care waste transportation problem[J]. *J. Comput. Sci.* https://doi.org/10.1016/j.jocs.2023.102104 (2023).
44. Sethanan, K. & Jamrus, T. Hybrid differential evolution algorithm and genetic operator for multi-trip vehicle routing problem with backhauls and heterogeneous fleet in the beverage logistics industry - *ScienceDirect[J] Computers Industrial Eng.*, 146. DOI:https://doi.org/10.1016/j.cie.2020.106571. (2020).
45. Walker, D. J., Everson, R. & Fieldsend, J. E. .Visualizing mutually nondominating solution sets in Many-Objective Optimization[J]. *IEEE Trans. Evol. Comput.* **17** (2), 165–184. https://doi.org/10.1109/TEVC.2012.2225064 (2013).

Acknowledgements

(Supporting) Research and development of intelligent optimization scheduling system for urban sanitation vehicles, project number: 4310436. This work was also supported by the Natural Science Research Project of Colleges and Universities in Jiangsu Province of China under Grant No. 24KJD120003, and the High-Level Talent Recruitment Program of Nanjing Xiaozhuang University (4172507), the Tianjin Municipal Education Commission Research Project (Grant No. 2024KJ072), 2023 Industry-University-Research Innovation Fund for Chinese Universities—New Generation Information Technology Innovation Project (no. 2023 IT 234), the Scientific Research Initiation Project of Tianjin University of Technology and Education (no. KYQD202332), and the Special Project for the Construction of Artificial Intelligence Courses (no. ZXY2024-12).

Author contributions
Zhang Haifei：Conceptualization, Software, Formal analysis, Investigation, Data Curation, Writing - Original Draft, Visualization, Funding acquisitionZhang Yuzhou：Methodology, Validation, Investigation, Resources, Writing - Review & Editing, Supervision, Funding acquisitionZhao Fen: Visualization, Modification, Writing - Review & EditingZhou Lujie: Modification, Writing - Review & Editing, Funding acquisitionZhou Bailing: Software, Data Curation, Funding acquisition.

Declarations

Competing interests
The authors declare no competing interests.

Additional information
Correspondence and requests for materials should be addressed to H.Z.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.