

A two-stage algorithm for bi-objective logistics model of cash-in-transit vehicle routing problems with economic and environmental optimization based on real-time traffic data

Yuanzhi Jin^a, Xianlong Ge^{a,*}, Long Zhang^b, Jingzheng Ren^{b,**}

^a School of Economics and Management, Chongqing Jiaotong University, Chongqing 400074, China

^b Department of Industrial and Systems Engineering, The Hong Kong Polytechnic University, Hong Kong, China

Corresponding author:

School of Economics and Management, Chongqing Jiaotong University, Chongqing 400074, China

Email: gexianlong@cqjtu.edu.cn (XL. Ge)

Department of Industrial and Systems Engineering, The Hong Kong Polytechnic University, Hong Kong SAR, China

Email: jzhren@polyu.edu.hk; renjingzheng123321@163.com (J.Z. Ren)

Abstract: Traffic congestion problems are very common in the large municipalities, especially in areas with karst features. Traffic jams happen in many key traffic nodes (such as the bridges across the river and the tunnels through the mountains) frequently, which may lead to severe challenges for the vehicle routing optimization. To effectively solve the routing problem of Cash-in-Transit (CIT) sectors, this study has aims to establish a novel bi-objective Cash-in-Transit Vehicle Routing Problem (CTVRP) model, including both the economic and environmental objectives, and design a Nearest Neighbor-first Iterated Local Search-second algorithm (NN-ILS) with the consideration of the special terrain. Then, a case study of a CIT company is performed to demonstrate the model and algorithm and a vivid solution is presented in real road network after the optimization by using the route fitting procedure. Meanwhile, the accuracy and effectiveness of the algorithm is verified by comparing it with several classical algorithms and OR-Tools solver. The experimental results show that the developed algorithm can help the decision-makers to obtain the solutions with high quality compared with the classical algorithms. Finally, the uncertainty of the developed algorithm is analyzed empirically and the Multi-Attribute Decision Making (MADM) combined with Principal Component Analysis (PCA) is utilized to support decision-makers to select the best satisfying solution instead of choosing the solution with minimum objective value(s).

Keywords: Cash-in-transit; Vehicle Routing Problem (VRP); Nearest neighbor algorithm; Iterated local search algorithm; Multi-attribute decision making

1 Introduction

The demand for financial services has significantly increased with the rapid development of China's economy. As the main body of the modern financial system, the banking system plays an important role in the steady growth, structural adjustment, and sustainable development of the economy. According to the 2019 China Banking Services Report released by the China Banking Association, there are 228,000 service nodes and 1,093,500 self-service devices deployed across the country by the end of 2019 in China. There are a large number of nodes and equipment to which a large amount of cash and other valuable bills are needed to be delivered or picked up every day, which leads to a huge challenge for CIT sectors.

To deal with such challenges, there are various have proposed models, algorithms, and risk assessment methods that have been developed for CTVRP [1-3]. However, there are still three research gaps in the previous work for the CTVRP: (1) The existing models and algorithms for the CTVRP are developed only for general problems and scenarios with general situations, and they may be inefficient for the CTVRP with specific geographical environments (such as the urban areas with mountains, rivers, or straits, etc.); (2) The environmental impact caused by transportation vehicles is overlooked, including CO₂ emissions and other pollutant emissions, as it is reported by the International Energy Agency (IEA) that transportation directly contributed to about 8040 million tons of carbon dioxide equivalent in 2017, accounting to 24.48% of the greenhouse gas emissions worldwide [4]; and (3) The distribution plans of the existing work only contain the visiting sequence of nodes, but there are still two questions that need to be answered by the logistics enterprise: (a) how to carry out the plans? and (b) Can it be completed within the time windows of customers? The above-mentioned research gaps motivated this study to develop a two-stage algorithm for bi-objective logistics model of cash-in-transit vehicle routing problems with economic and environmental optimization simultaneously based on the real-time traffic data.

In this work, we focus on the modeling and treatment of a real-life application related to

CTVRPs. Firstly, a bi-objective logistics model considering the travel distance and the emissions of CO₂ and other major pollutants is established for CTVRPs. Then, a two-stage algorithm based on the spatial clustering algorithm is proposed by using real-time traffic information for real scenarios with specific geographical environments. In addition, the MADM method is employed to objectively evaluate the feasible solutions obtained by the proposed algorithm, considering multiple attributes in security, sustainability, and comfort aspects. Finally, a route fitting procedure is developed to guide remote vehicles online, and the information of detailed path with the lowest emissions between two adjacent nodes is sent to the remote vehicle based on the real-time traffic state provided by an online map. Besides the aforementioned factors, the risk index of the routes should also be considered because the onboard goods are cash and other valuables. Although some new security devices and technologies have been widely equipped, such as modified and reinforced vehicles, weapons on board, and real-time GPS tracking [5], etc., to deal with the potential risk of robbery, robbery incidents of armored vehicles occur occasionally all over the world [1]. This paper aims to generate stochastic routes directly to reduce the risk of robbery in the routing optimization phase.

The remaining parts of this paper are organized as follows. Section 2 presents a comprehensive literature review for identifying the research gaps and highlight the scientific contributions of this study. Section 3 introduces the routine tasks of CIT sectors as well as a novel route planning idea under the special terrain. In Section 4, a mathematical model of CTVRP is described, and the detailed implementation of the NN-ILS algorithm for solving the model is proposed in Section 5. A case study is conducted to demonstrate the model and the algorithm in Section 6. Finally, this study has been concluded in Section 7.

2 Literature review

The Vehicle Routing Problem with Time Windows (VRPTW) is a variant of classic VRP. Originally, the general model of VRPTW was proposed by Solomon [6] in 1987, and then the related heuristic, metaheuristic, and exact algorithms have been well developed [7]. Prins [8] introduced a

new merge heuristic based on the savings heuristic [9]. A large-scale real case of multi-trip VRP was solved by the heuristic for minimizing the total cost and the number of required vehicles. Pureza et al. [10] presented a model of VRPTW with multiple deliverymen to deal with the distribution of beverages and tobacco in the urban areas. Two algorithms, a tabu search-based algorithm and an Ant Colony Optimization (ACO) algorithm, were developed for obtaining the solutions with the minimum cost. In another study, Zhang et al. [11] developed a multi-objective model of VRP with flexible time windows, for minimizing the total distribution costs and maximizing the overall customer satisfaction. A new Pareto-based multi-objective ACO algorithm using mutation operators was designed, and it was employed on the well-known benchmark Solomon's problems and a real case. Later, Salavati-Khoshghalb et al. [12] examined the VRP with stochastic demand and designed two recourse policies, including a return and refill the capacity and a return in advance according to the residual capacity, when the failure events occurred. An exact algorithm framework integrating the Integer L-shaped algorithm into a branch-and-cut algorithm is proposed for solving the uncertain problem. Pasha et al. [13] proposed a novel concept of the "factory-in-a-box" planning problem, which included an assessment sub-problem and a transportation sub-problem between suppliers and manufacturers. To solve the transportation sub-problem, four metaheuristic algorithms, including the Genetic Algorithm (GA), Variable Neighborhood Search (VNS), Tabu Search (TS), and Simulated Annealing (SA), were employed for large-scale instances. Recently, the latest metaheuristic, namely the adaptive memory social engineering optimizer, was proposed for the sustainable vehicle routing problem [14]. The problem framework included three objectives that related to financial, environmental, and social considerations respectively. The work mentioned above provides a variety of general methods for solving the VRPTW.

The CTVRP is derived from the traditional VRPTW, but it has its unique characteristics. Since CIT vehicles carry cash, gold, jewelry, and other valuables, many studies considering the risk of being robbed were carried out. Tarantilis and Kiranoudis [15] developed a GPS-based decision support

system that measured the risk as a function of the distance from a customer to the nearest police station and designed an adaptive memory programming method to solve the CVRP model. Talarico et al. [1] proposed a Risk-Constrained Cash-in-Transit Vehicle Routing Problem (RCTVRP), and defined the risk of the route as the function related to the probability of a robbery happening, probability of a successful robbery and the quantified loss after the robbery happens. Subsequently, Talarico et al. [16] further simplified the risk calculation method for cash pick-up, a cumulative function that was positively correlated with the travel distance and the amount of cash on board. As for the solution method, the authors adopted an ACO-LNS algorithm (i.e., ACO and Large Neighborhood Search (LNS) with seven operators) to find the feasible solutions. To solve the RCTVRP, Radojicic et al. [2] designed a greedy randomized adaptive search procedure hybridized with path relinking methodology and constructed a new data structure to reduce the time complexity. Although the studies mentioned above have achieved fruitful results for the CTVRP, they did not consider the actual road traffic, especially the traffic network within the city. Therefore, developing a model which can incorporate the real-time traffic information in urban areas is prerequisite.

There are some studies considering the uncertainty of travel time between nodes when solving CTVRP. Chang [17] proposed a CTVRP model with stochastic travel time to formulate the variant distribution plans and to reduce the risk of robbery by using the time-space network flow technique. In the same context, Yan et al. [18] established a RCTVRP model by using the time-space network technology. A mathematical programming software and decomposition/collapsing technology were employed to solve the model. Boonsam et al. [19] studied assignment problems and VRPTW, taking Bangkok bank, Thailand as an example. Three heuristic algorithms were used to address the problems, aiming to improve the distribution efficiency by utilizing existing resources. Sabar et al. [20] proposed a Self-Adaptive parallel Evolutionary Algorithm (SAEA) to handle the dynamic vehicle routing problems considering traffic congestion. Although these studies have considered the influence of road traffic conditions on travel time and established corresponding dynamic models, the

unified models are not necessarily suitable for each specific road segment in the real scenario. Therefore, it is unrealistic to design a unified model for traffic congestion to suit route segments in all the cities. Aiming at the perception of real-time traffic information related to the distribution routes, an additional route fitting procedure should be used during the distribution process.

Many hybrid or multi-stage algorithms require an initial solution. The Nearest Neighbor (NN) algorithm [6] and Clarke and Wright's savings method [9] are commonly used in neighbor search-based algorithms. Meanwhile, the First Come First Served (FCFS) policy and its variations are more suitable for population-based algorithms [21]. Küçüköglü and Öztürk [22] designed a hybrid meta-heuristic algorithm for solving a Vehicle Routing Problem with Backhauls and Time Windows (VRPBTW), in which the improved NN algorithm was employed to obtain an initial solution. Based on this, the NN algorithm has been extended to some new algorithms such as the k-NN algorithm [23] and multi-purpose and multivariable weighted nearest neighbor algorithm [24], etc. In addition, the savings algorithm has a higher time complexity than the NN algorithm, which is not conducive to solving large-scale instances in a short time. Therefore, the nearest neighbor algorithm is adopted in this paper to obtain an initial solution. However, the quality of the initial solution is generally bad and the local search algorithm [25] is very suitable for its intensification, which generates new neighbor solutions by performing various neighborhood transformations on the initial solution to achieve optimization. Over the past 20 years, many variations of local search algorithms have been developed according to different neighborhood transformation methods, such as multi-start local search algorithm for VRPTW [26], Green Vehicle Routing Problem (GVRP) [27], and stochastic local search algorithm [28], etc.

Generally speaking, the Iterative Local Search (ILS) is one of the explorative local search methods, adding perturbation to the local optimal solution obtained by local search and repeating it again and again [29]. Because of its good performance, the ILS has been widely used in routing optimization recently, such as the CVRP [30] and open VRPTW [31]. Different from the literature re-

viewed above, the termination condition of iterative local search adopted in this paper is a time limit instead of certain iterations, which is more flexible for addressing the instance in different scale.

Many algorithms, such as the ACO [32, 33], SA [34] and TS [35], designed for combinatorial optimization problems have been widely used in the logistics industry during the past two decades. Similar to the work of Pasha et al. [13], these algorithms and the well-known OR-Tools solver [36] are also employed to verify the performance of the NN-ILS algorithm proposed in this paper. OR-Tools is an excellent combinatorial optimization solver; for certain Travelling Salesman Problem (TSP) instances with millions of nodes, solutions have been found by the solver guaranteed to be within 1% of an optimal tour. In recent years, scholars have proposed a new framework based on reinforcement learning to solve the TSP [37], VRP [38], Prize Collecting TSP [39] and other variants of VRPs, in their experiments, and OR-Tools solver was used as a baseline.

The research framework of this paper is based on the work of Talarico et al. [1], and there are four aspects of innovations: (1) The constraint of time windows is added to make the model more practical; (2) trying to ensure a balance of the effective working hours in the same business district; (3) Baidu Map (an online map provider, similar to the Google Map) is used to obtain the real-time traffic information to effectively avoid congested road sections; and (4) The theory of multi-attribute decision making is utilized to select the most satisfactory solution. Meanwhile, this paper has the following four-fold scientific contributions:

- (1) Formulating a CTVRP model and a solution framework considering the special terrain;
- (2) Designing a NN-ILS algorithm to solve the CTVRP with time windows;
- (3) Enriching the adaptiveness to changeable traffic flow and display of cash transport scheme via the designed route fitting procedure; and
- (4) Applying the MADM for solution selection.

3 Business description and optimization idea

In this section, the daily tasks of the CIT sectors are described briefly. Combined with the commercial distribution and traffic network characteristics of cities with special terrain, the idea of optimization algorithm for CIT sectors is presented.

3.1 Introduction of CIT business

CIT sectors usually provide cash transport services from the branches to a central bank. The sectors with a limited number of armored vehicles need to deliver or pick-up cash timely for each branch every day. The vehicles deliver the cash box pairs (including two boxes, one for cash and another for bills) from a central bank to each branch every morning, and the pick-up procedure is reversed in the afternoon. The cash containers to be transported from or to each branch are marked by code, and the demand of each branch can be held in only one box pairs per day. Additionally, each vehicle can hold up to 30 boxes, i.e., serve at most 15 branches.

3.2 Routing optimization for the cities with special terrain

Influenced by the natural topography, some cities are built along the mountains or rivers, which are called mountain cities. It is hard to build cities in areas with natural terrains such as mountains, hills, and rivers, etc. Instead, mountainous cities are formed in clusters combined with local terrain conditions. However, it has the special geographical environment that causes many inconveniences in the commute of citizens. To solve this tension, governments have built a multitude of first-class tunnels and bridges in urban transportation network. Over time, several commercial clusters (hereinafter referred to as business districts) are formed in a few flat areas between the mountains. Then, the bridges and tunnels are the bottlenecks of the roads connecting different business districts. Because the public has huge travel demand in the same time interval, the bottlenecks connecting various business districts are congested for long time during rush hours even all day. At the same time, low visibility caused by frequent dense fog in the mountain city and the steep slopes on many roads has

led to a higher possibility traffic accident, which is also a severe challenge for the urban logistics industry.

The order-first split-second method has been widely used to solve the CVRP model [40]. In the first stage, a giant TSP tour is obtained, and then the tour is split by the sweeping method or greedy method etc. according to the location of the depot and customers to achieve a VRP solution (see Fig. 1).

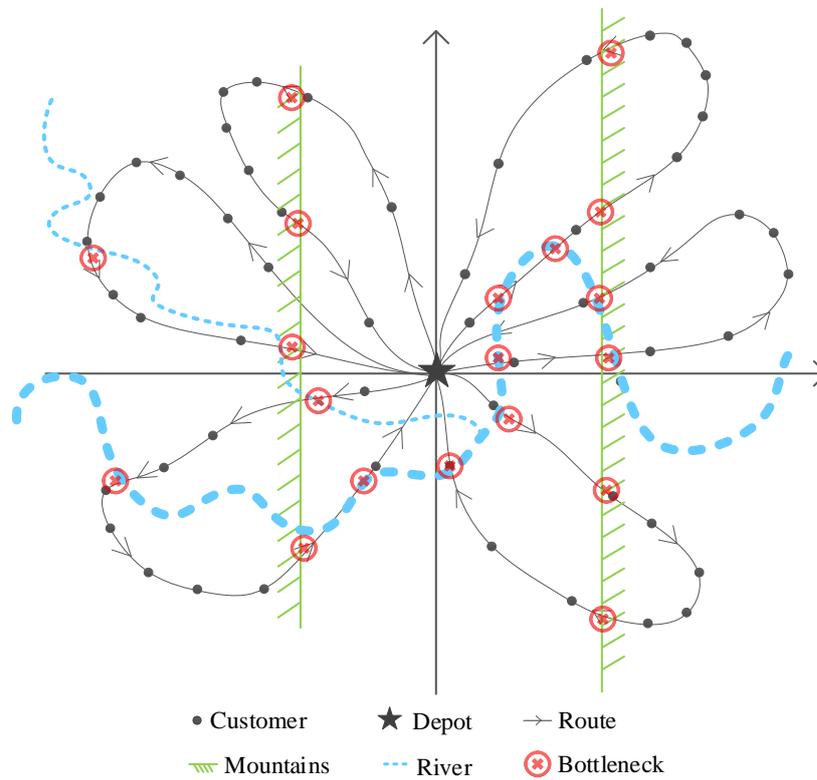


Fig. 1. A solution results from traditional methods.

However, this method does not consider the bridges and tunnels in real road networks which are prone to experiencing traffic jams. It is possible that the delivery vehicles are still moving slowly on the bridges or in the tunnels when the branches' opening time has passed if the method is applied to generate the routes for CIT sectors. Integrated with the actual driving experience of the drivers in mountainous cities, we proposed a new routing optimization idea based on the business districts to solve this problem (see Fig. 2).

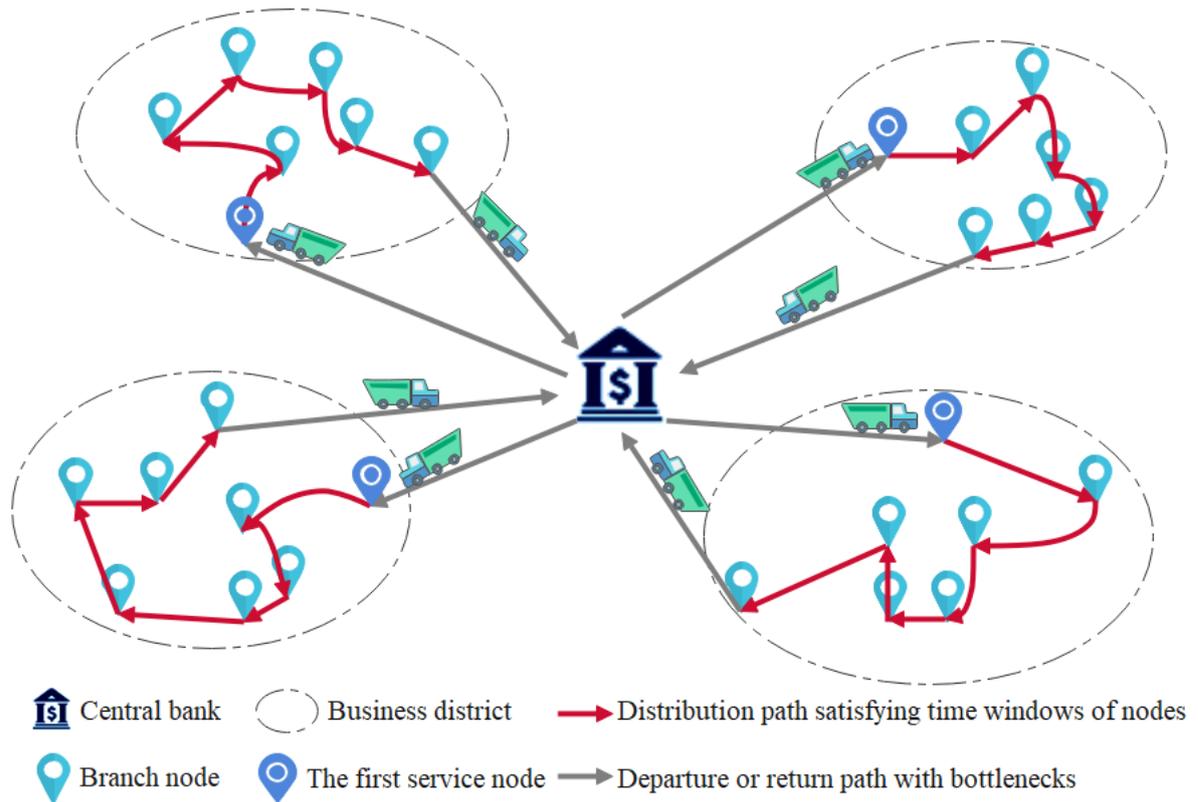


Fig. 2. A new idea for routing problems based on the mountainous city.

Depending on the characteristics of road traffic, each CIT vehicle is required to leave from the central bank in advance, crossing congestion-prone bridges or tunnels before the arrival of traffic jams. Then, ensure that all vehicles reach the first serviced node before the time window opens, and arrive at the last node they serve before the time window ends. In the next section, the nomenclatures used in the model presented in this paper are listed and the mathematical model is established.

4 The CTVRP model

The CTVRP model studied in this paper belongs to a special VRPTW, which has the characteristic that all branches have the unit demand and the uniform time window. The over-concentrated time windows bring great challenges to the efficient dispatching of vehicles. In addition, under the mountainous urban environment, the bank branches are mainly distributed in various business districts, and the distance between the business districts is relatively far. Therefore, a mathematical

model based on business districts is established for the CTVRP. The parameters and variables used in this model were presented in Table 1.

Table 1

Nomenclature used in the CTVRP model

Model Component		Meaning
Type	Nomenclature	
Sets	$C = \{2, 3, \dots, n+1\}$	the set of customers (indexed by i, j)
	$V = \{1\} \cup C$	the set of nodes (indexed by i, j)
	$B = \{1, 2, \dots, b\}$	the set of business districts (indexed by l)
	$K = \{1, 2, \dots, m\}$	the set of vehicles (indexed by k)
	$S = \{1, 2, \dots, s\}$	the set of candidate depots (indexed by i)
Decision variables	$x_{i,j}^k$	1 if vehicle k travels the $arc(i, j)$; 0 otherwise
	y_i^k	1 if customer i is served by vehicle k ; 0 otherwise
	s_i	1 if the candidate i is selected as the depot; 0 otherwise
	Z_i^b	1 if customer i belongs to the business district b ; 0 otherwise
Parameters	w_i	the service duration of customer i
	$t_{i,j}$	the travel time from node i to j
	$d_{i,j}$	the travel distance from node i to j
	$v_{i,j} = d_{i,j} \cdot (t_{i,j})^{-1}$	the average speed traveled from node i to j
	T	the efficient working time
	Q	the capacity of vehicles
	M	sufficiently large positive number
	\underline{U}	the minimum number of customers that the vehicle needs to serve
	SF	the average service fees paid by customers each year
	EYC	the average total cost of vehicles per year
	α	the profit factor of the vehicle
	m	the maximum number of vehicles available
EF	the emission factor of fuel	

Based on Table 1 and the previous description, a bi-objective CTVRP model for minimizing the total distance and the total emissions is established.

$$\text{Min } z_1 = \sum_{i \in V} \sum_{j \in V} \sum_{k \in K} d_{i,j} \cdot x_{i,j}^k \quad (1)$$

$$\text{Min } z_2 = EF \cdot \sum_{i \in V} \sum_{j \in V} \sum_{k \in K} d_{i,j} \cdot f_{FC}(v_{i,j}) \cdot x_{i,j}^k + \sum_{p \in \{1-4\}} \sum_{i \in V} \sum_{j \in V} \sum_{k \in K} d_{i,j} \cdot f_p(v_{i,j}) \cdot x_{i,j}^k \quad (2)$$

s.t.

$$\forall i \in C, k \in K: \sum_{j \in V} x_{i,j}^k = \sum_{j \in V} x_{j,i}^k = y_i^k \quad (3)$$

$$\forall i \in C: \sum_{k \in K} y_i^k = 1 \quad (4)$$

$$\underline{U} = \alpha [EYC / SF] \quad (5)$$

$$\forall k \in K: \sum_{i \in C} y_i^k \geq \underline{U} \quad (6)$$

$$\forall k \in K: \sum_{i \in C} q_i \cdot y_i^k \leq Q \quad (7)$$

$$\forall k \in K: \sum_{i \in C} \sum_{j \in C} t_{i,j} \cdot x_{i,j}^k + \sum_{i \in C} w_i \cdot y_i^k \leq T \quad (8)$$

$$\sum_{k \in K} \max_{j \in C} (y_j^k) \leq m \quad (9)$$

$$\forall i, j \in C, k \in K, l \in B: Z_i^l - (1 - x_{i,j}^k) \cdot M \leq Z_j^l \quad (10)$$

$$\sum_{i \in S} s_i = 1 \quad (11)$$

$$\forall k \in K, i \in S: s_i = \sum_{j \in C} x_{i,j}^k = \sum_{j \in C} x_{j,i}^k \quad (12)$$

$$\forall H \subseteq C, h \in H, k \in K: \sum_{i \in H} \sum_{j \in \{V \setminus H\}} x_{i,j}^k \geq y_h^k \quad (13)$$

$$x_{i,j}^k \in \{0,1\} \quad \forall i, j \in V, i \neq j, k \in K \quad (14)$$

$$y_i^k \in \{0,1\} \quad \forall i \in V, k \in K \quad (15)$$

The first objective z_l (1) is to minimize the total travel distance. Constraints (3) and (4) guarantee that each branch is served only once. Constraints (5) and (6) represent the minimum number of branches served per vehicle. Constraints (7) define the capacity constraint of the vehicles. Constraints (8) represent that the vehicles arrive at the branches within their time windows. Constraint (9) means that the number of vehicles used cannot exceed the number of the fleet owned. Constraints (10) represent that the branches served by a vehicle belong to the same cluster. Constraint (11) en-

sures that only one central bank is available. Constraints (12) guarantee that all vehicles must start from and return to the central bank. Finally, sub-tours are eliminated by constraints (13).

As for the environmental aspect, the second objective z_2 is to minimize the total emissions, including CO₂, CO, NO_x, VOC and PM. The first part of the objective function z_2 aims to calculate the total CO₂ emissions contributing by the energy consumption between two adjacent nodes and the second part represents the total emissions of the other four major pollutants. According to the Computer Programme to calculate Emissions from Road Transport (COPERT) model [41], the function $f_{FC}(v)$ is designed to calculate the speed dependency of fuel consumption factor with the unit g fuel.km⁻¹. Then, the CO₂ emission of the corresponding road segment can be obtained based on the CO₂ emission factor, the travel distance (unit km) and the EF (unit g CO₂. g fuel⁻¹). Similarly, the function $f_p(v)$ is designed to calculate the speed dependency of pollutant emission factors with unit g.km⁻¹. Each pollutant emission factor can be to calculate corresponding emission based on the travel distance (unit km).

In addition to the COPERT model, many similar models are developed to calculate emission factors. For example, the European Commission's Methodologies for Estimating air pollutant Emissions from Transport (MEET), the UK's National Emissions Inventory (NAEI) and the Comprehensive Modal Emission Model (CMEM) focusing on the emissions and fuel consumption of heavy diesel trucks [42]. The COPERT model is used in this study to calculate the emission factors of CO₂ and other regulated pollutants, because the raw data is obtained from the real-world intelligence traffic information system and some new technologies such as data mining, machine learning are employed to obtain the regression functions for hundreds of different vehicles.

The CIT trucks used in the case study have a gross weight of 3.51 tons, belonging to the smallest diesel Heavy-Duty Vehicles (HDV, 3.5t<gross weight<7.5t) defined in the COPERT model. However, the weight of 3.5t is the boundary between the Light-Duty Vehicles (LDV, gross weight<3.5t) and HDVs, and these two categories have different speed-dependent (variable v) re-

gression functions. The regression functions defined for the HDVs are [41]:

$$f_{FC}(v)=\begin{cases} 1425.2v^{(-0.7593)} , 0 < v < 47 \\ 0.0082v^2 - 0.043v + 60.12 , 47 < v < 100 \end{cases} \quad (16)$$

$$f_p(v)=\begin{cases} 37.28v^{(-0.6945)} , p = CO, 0 < v < 100 \\ 50.305v^{(-0.7708)} , p = NO_x, 0 < v < 46.7 \\ 0.0014v^2 - 0.1737v + 7.5506 , p = NO_x, 46.7 < v < 100 \\ 40.12v^{(-0.8774)} , p = VOC, 0 < v < 100 \\ 4.5563v^{(-0.707)} , p = PM, 0 < v < 100 \end{cases} \quad (17)$$

The regression functions defined for the LDVs are [41]:

$$f_{FC}(v)=0.0198v^2 - 2.506v + 137.42 \quad (18)$$

$$f_p(v)=\begin{cases} 22.3*10^{-5}v^2 - 0.026v + 1.076 , p = CO \\ 24.1*10^{-5}v^2 - 0.03181v + 2.0247 , p = NO_x \\ 1.75*10^{-5}v^2 - 0.00284v + 0.2162 , p = VOC \\ 4.5*10^{-5}v^2 - 0.004885v + 0.1932 , p = PM \end{cases} \quad (19)$$

Similar to the calculation method of CO₂ emission in [43], the COPERT model takes into account not only the total weight and the travel distance of the vehicle, but also the average speed. In order to determine which group of regression functions is more suitable for calculating the emissions of CIT vehicles, the refueling record and historical mileage of vehicles were analyzed, and it could be concluded that the average fuel consumption of vehicles is about 12 litres·100km⁻¹. Equation (20) can be used to roughly calculate empirical CO₂ emissions:

$$E_{co_2}(d)=EF * d * FC * \rho \quad (20)$$

where FC is fuel consumption with travel distance d , and ρ is the density of diesel. Since carbon dioxide contributes to most of the total emissions, other pollutants can be ignored for rough comparisons. After one test, the results of calculating the emissions between the LDV regression function and the empirical carbon emissions are approximate, while the result obtained by the HDV regression functions is about 57% higher than the empirical method obtained. There are two possible reasons for this issue: (1) the gross weight of the CIT vehicles includes the weight of six crewmembers;

however, in fact, each vehicle only has four crewmembers at operation time; (2) the cash boxes transported by CIT vehicles are relatively light comparing with the conventional goods. Therefore, the regression functions in Eqs. (18) and (19) are used to calculate the total emissions.

The CPLEX solver (academic edition: 12.10) is employed to solve the small-scale instances randomly generated. The optimal solution can be obtained in a few seconds if the number of customers is less than 18; however, the solver becomes powerless when facing the instances with $n > 18$ because constraints (13) contain an exponential operation to enumerate all subsets of the set C . Therefore, the NN-ILS algorithm is designed for large-scale instances to solve this problem in the next section.

5 Algorithms for the CTVRP

The CTVRP studied in this paper is equivalent to the VRPTW with unit demand and uniform time windows. There are many existing algorithms mentioned in the literature review that can solve this problem directly. However, when facing the distribution area with special terrain (i.e., traffic network bottlenecks exist scatteredly), the distribution scheme obtained by these algorithms may not be completed on time. To solve this tension, this study designs a NN-ILS algorithm framework considering the special terrain, which can make the solution more suitable. Firstly, the *K-means* algorithm [44] is utilized to cluster all customers (branches) according to their geographical location. Each cluster represents a business district and geographical closeness among customers may be classified into the same district. In addition, the optimal value of K is obtained by the Elbow Method [45]. Then, an improved nearest neighbor algorithm is performed to obtain an initial TSP tour in each district, and a time-limited iterated local search algorithm is employed to further optimize the TSP tours. Following that, the optimized TSP tours are split according to the time window constraint and the capacity constraints. Eventually, a VRP solution is obtained by collecting the output routes in each cluster. The whole process of the proposed NN-ILS algorithm is illustrated in Fig. 3.

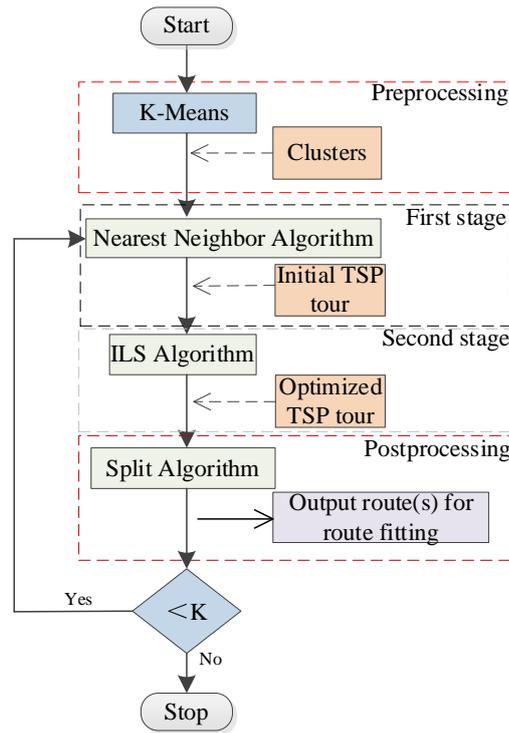


Fig. 3. An overview of the NN-ILS algorithm.

5.1 Nearest neighbor algorithm

The NN algorithm is an effective constructive heuristic [6], which has the advantages of low time consumption, and ease to be understood and programmed. The algorithm starts every route by finding the unassigned branch "closest" to the central bank. Then, the heuristic searches for the branch "closest" to the newly added route. As soon as the search fails (making the route infeasible after adding any unrouted branch), a new route will start, unless there are no more branches to schedule. Finally, an initial VRP solution is obtained after the algorithm stops. Thus, we can observe that the basic idea of the NN algorithm is similar to the greedy algorithm [46].

The most important characteristic of the NN algorithm is that it is simple and can obtain an initial solution in a short time. However, it has fast convergence and is prone to trap in local optimum. The time window and capacity constraints are first relaxed and the NN algorithm is only employed to generate an initial TSP tour inside each business district. Since there are multiple choices for the location of the depot, the NN algorithm needs to be modified before using it in this work. Therefore,

this paper used each branch as the starting point of the route construction within the business district to generate multiple TSP tours, and finally chooses the TSP tour with minimum cost as the initial tour within each cluster. The modified nearest neighbor algorithm is described in **Algorithm 1**.

Algorithm 1 Pseudo-code for the modified NN algorithm

```

1: Input: the nodes in a business district
2: Calculating a distance matrix between each pairs of nodes
3: Related parameters: nearest-neighbor (nn), route record list (RRL), current route (CR).
4: Initialize counter: iteration=0
5: While iteration<nodeNumber
6:   startPoint = iteration, unvisited = startPoint
7:   Add startPoint to CR
8:   While unvisited != null
9:     Search a nearest-neighbor nn in unvisited according to the distance matrix
10:    Add nn to CR
11:    Delete nn from unvisited
12:   End while
13:   Add CR to RRL
14:   Clear CR
15:   iteration = iteration + 1
16: End While
17: Output min(RRL)

```

5.2 A time-limited iterated local search algorithm

Under the framework of the original ILS algorithm [47], a perturbation operator and a control parameter of running time are added to make the algorithm more accurate and flexible. Taking the result of the nearest neighbor algorithm as an input, the time-limited iterative local search algorithm is used to further optimize an initial TSP tour. To speed up the search process, two kinds of search operators: *2-opt** exchange (to avoid confusion with the well-known *2-opt* operator based on arc-exchanging [48], a different notation is used herein) and *n-opt* random permutation operator are designed. The function of the former is mainly to exchange any node pairs for getting a better neighbor, while the latter can generate a large disturbance to the current solution by setting a proper value *n*, thus it may help the algorithm to escape from local optimum. In general, the value of *n* should not be too large, which will increase the time complexity of the algorithm ($O(n!)$).

In each iteration, the *2-opt** operator is first used to perform pairwise exchanges among all

nodes to find the best exchange; then the n -opt operator is employed to randomly perturb the current solution. In this paper, a timer is adopted to control the time consumption of the iterations; that is, the outer loop is forced to terminate when the timer reaches the preset value and the current optimal solution is outputted simultaneously. The pseudo-code of the time-limited iterative local search algorithm is shown in **Algorithm 2**.

Algorithm 2 Pseudo-code for the time-limited ILS

```

1: Input X: the output of the NN
3: Set the parameters: history optimal solution
   (best), current solution (current).
4: Initialize timer  $t_0$ .
5: While True
6:    $X^* \leftarrow$  Perform 2-opt* on X
7:    $current \leftarrow$  Length ( $X^*$ )
8:   If  $current < best$  Then
9:      $best = current$ 
10:    Record  $X^*$ 
11:   End If
12:    $X \leftarrow$  Perform  $n$ -opt on  $X^*$ 
13:   If  $currentTime - t_0 > cutOffTime$  Then
14:     break and goto 17
15:   End If
16: End While
17: Output best solution

```

5.3 Splitting method

After obtaining the optimized TSP tour in each business district, it may be split to satisfy the time window constraint of all customers and balance the working time between the routes within the same district. It is virtually impossible for a vehicle to serve 15 customers if the width of the time window is 120 minutes because of a fixed service time of 3 minutes per customer and an average travel time of 8 minutes between adjacent customers; therefore, we can ignore the capacity constraint when splitting the TSP tours.

We assume that the TSP tour $[b_0, b_1, b_2, \dots, b_{n-1}, b_0]$ inside a business district and the travel time between adjacent nodes $\{t_1, t_2, \dots, t_{n-1}, t_n\}$ can be obtained from Baidu Map in real-time. Then the total travel time is easily calculated. Since the delivery time in the morning is valuable, the path pref-

erence selected is “shortest time consumption” (avoid congestion automatically) when requesting data from Baidu Map. However, this could increase the travel distance by around 10 percent compared with other preferences selected. The potential increase in travel distance seems contradictory to objective z_1 ; however, the vehicles may serve as many customers as possible and bring more revenue to CIT sectors because of the time saving. The total travel time plus the total service time of all customers within the business district is the total time consumption, denoted as TBD . Consequently, the number of vehicles needed by the business district can be evaluated by:

$$KBD = \lceil TBD / T \rceil \quad (21)$$

Finally, the sweep method is employed to spilt TBD into KBD segments (see Fig. 4a). Obviously, multiple TSP tours are obtained after inserting the depot into each segment. A complete VRP solution could be obtained after processing all TSP tours of the business district.

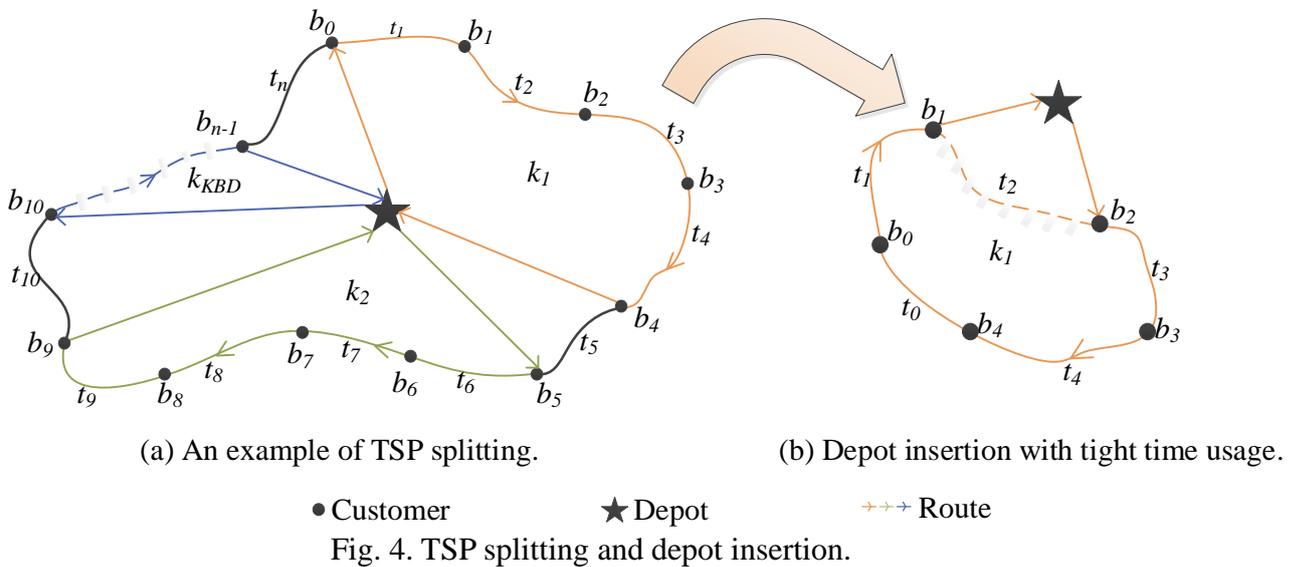


Fig. 4. TSP splitting and depot insertion.

Taking k_1 as an example, as is shown in Fig.4b. In order to make full use of the working time T after connecting b_4 to b_0 , the arc (b_1, b_2) with the longest travel time t_2 is selected to disconnect, and then the depot is inserted between b_1 and b_2 . Finally, the route $[DC, b_2, b_3, b_4, b_0, b_1, DC]$ is completely constructed.

In Eq.21, "[*]" represents the integer operation with the two options—*Ceiling* or *Floor*; the

former ensures that each vehicle can complete its tasks within effective working time in a business district while the latter cannot ensure that the delivery tasks can be completed on time. Fortunately, the average speed of CIT vehicles is faster than that of private cars that are the main traffic participants in the urban areas and the travel time between each pair of nodes obtained from the Baidu Map is based on the speed of private cars; meanwhile, the vehicles are required to arrive at the first customer they serve at the beginning of the time window. These two aspects make it possible for the vehicles to complete their tasks on time when the latter is used. Of course, one vehicle assigned is not enough in any way under an extreme condition (e.g., $T=120$ minutes and $TBD =239$ minutes); the *Floor* operation can be adopted only when there are more vehicles needed in a business district, and it means that “the remainder customers” are allocated to other routes equally.

Finally, the use of the *Ceiling* or the *Floor* depends on the profit preference of the decision-makers. The conservative way is to choose the *Ceiling*, which can not only improve the Quality of Service (QoS), but also relieve their staff's work pressure. Otherwise, CIT sectors may get a higher revenue with fewer vehicles.

6 Case study

A CIT company in the mountainous city Chongqing, located in southwest of China, is used as the case study to demonstrate the developed model. The NN-ILS algorithm proposed in this paper is employed to generate the tour plan, and then the route fitting procedure and risk assessment method are used and the impact of depot location on cost is analyzed. Meanwhile, the performance of the proposed algorithm is verified by compared with other algorithms. Finally, the uncertainty analysis on the proposed algorithm and the application of MADM for ranking solutions are presented in section 6.3.

6.1 The case study based on objective z_1

The company provides cash transport services to a bank that includes 146 branches (or custom-

ers, numbered B-1 to B-146). Due to a non-disclosure agreement signed with the company, the locations of the customers are randomly modified. In fact, the company also serves several relatively remote and rural branches, which are additionally optimized by using the open vehicle routing problem model [49].

6.1.1 Tour plan

The opening time of each branch is 9:00 am and the earliest ready time is 7:00 am, so the effective working time of the CIT vehicles is two hours (i.e., $T=120$ minutes) during the morning delivery process. In order to serve more customers and avoid congestion, the early departure strategy is employed. The company requires each vehicle to arrive at the first customer they served before 7:00 am. Similarly, the effective working time is also two hours during the afternoon pick-up process and may be extended because of the heavy traffic in the rush hours.

According to the survey for the company, the fuel consumption of the vehicle is 12 liters per 100 kilometers, assuming that the price of diesel is 6.28 (Yuan RMB) per liter and the total salary of four employees for each vehicle is 800 (Yuan RMB) per day. The main parameters used in the algorithm are presented in Table 2.

Table 2

Parameters used in the case study.

Type	Parameters	Description	Values
Vehicle	V_c	fuel cost per km	0.754 (Yuan RMB)
	F_c	fixed cost of vehicles	800 (Yuan RMB)
	$oilPrice$	the price of diesel	6.28 (Yuan RMB).L ⁻¹
ILS	$cutOffTime$	the value of time limited	2s
	n	the number of nodes being perturbed in $n-opt$	4

Theoretically, the best location of the depot (denoted as DC0) is the core of clustering centers. Therefore, the DC0 is used as the depot to further obtain the whole solution. The best solution is obtained after running three times of the proposed NN-ILS algorithm, as illustrated in Fig.5, and the detailed node sequence of each route is listed in Table 3.

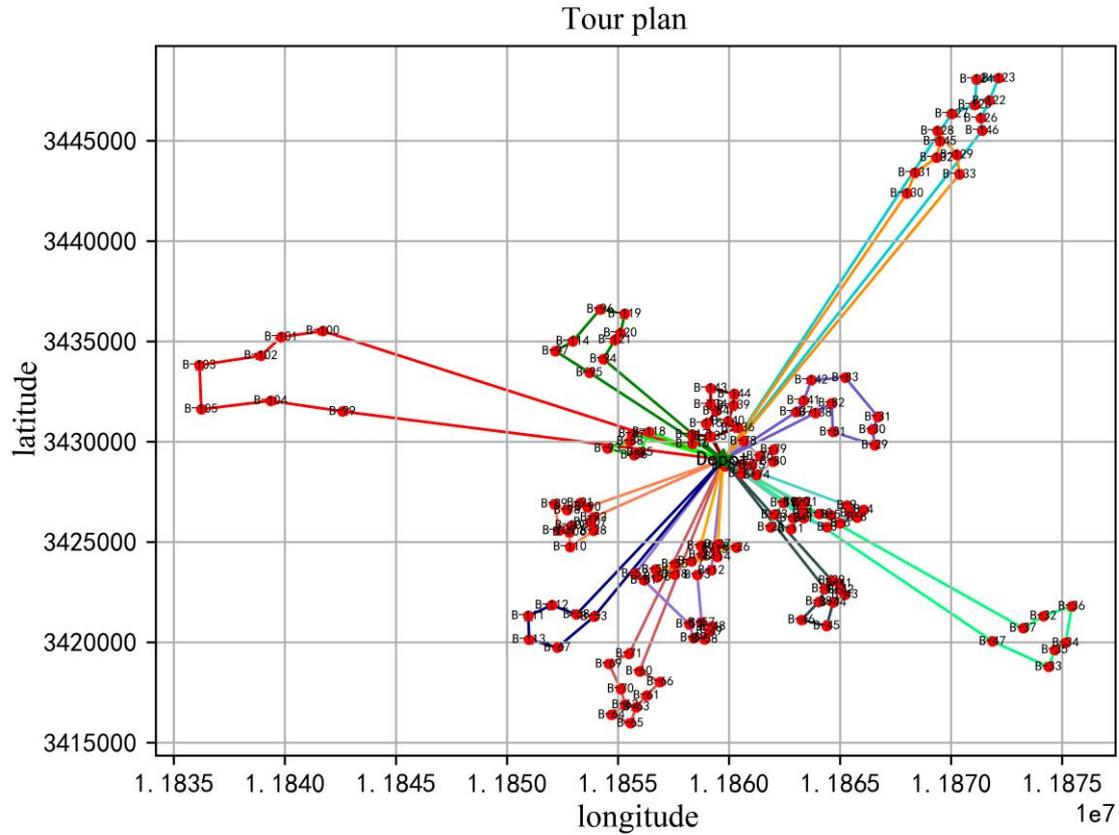


Fig. 5. A brief tour plan for minimizing z_I .

Table 3

The visiting sequence of the tour plan.

NO.	Visit order
R01	DC0->B-72->B-73->B-77->B-75->B-74->B-80->B-79->B-76->B-78->DC0
R02	DC0->B-135->B-117->B-116->B-136->B-140->B-139->B-144->B-143->B-134->B-84->B-115->DC0
R03	DC0->B-127->B-125->B-124->B-123->B-122->B-126->B-146->DC0
R04	DC0->B-71->B-69->B-70->B-62->B-64->B-65->B-63->B-61->B-66->B-60->DC0
R05	DC0->B-99->B-104->B-105->B-103->B-102->B-101->B-100->DC0
R06	DC0->B-47->B-33->B-35->B-34->B-36->B-32->B-37->DC0
R07	DC0->B-12->B-13->B-57->B-48->B-49->B-58->B-59->B-50->B-51->B-52->DC0
R08	DC0->B-27->B-26->B-15->B-14->B-54->B-56->B-18->B-55->B-17->B-22->B-16->DC0
R09	DC0->B-92->B-107->B-28->B-110->B-106->B-108->B-109->B-89->B-98->B-91->B-90->DC0
R10	DC0->B-118->B-85->B-86->B-93->B-88->B-87->DC0
R11	DC0->B-39->B-41->B-42->B-43->B-44->B-45->B-46->B-38->B-40->DC0
R12	DC0->B-1->B-3->B-6->B-2->B-8->B-4->B-9->DC0
R13	DC0->B-19->B-23->B-25->B-11->B-24->B-5->B-10->B-7->B-21->B-20->DC0

R14	DC0->B-94->B-121->B-120->B-119->B-96->B-114->B-97->B-95->DC0
R15	DC0->B-130->B-131->B-132->B-145->B-128->B-129->B-133->DC0
R16	DC0->B-137->B-141->B-142->B-83->B-31->B-30->B-29->B-81->B-82->B-138->DC0
R17	DC0->B-53->B-67->B-113->B-111->B-112->B-68->DC0

Although the routes shown in Fig. 5 can basically satisfy the needs of the decision-makers, they are not really practical for the drivers because it is only a sketch map. In addition, the traffic conditions between any adjacent nodes may change during the distribution process. Accordingly, the CIT vehicles need to be guided to avoid congestion. Therefore, a route fitting procedure is designed based on the tour plan. The procedure uses multithreading technology to guide each remote vehicle, one thread for a vehicle. There are multiple paths to choose between two adjacent nodes in urban areas. When the vehicle arrives at a customer node, the corresponding thread requests real driving paths between the current node and the next node from Baidu Map, and then selects the least time-consuming path from the returned results. Finally, detailed information of the path is sent to the remote vehicle. The corresponding thread ends until the vehicle returns to the depot. The detailed routes between all adjacent nodes are fitted into the actual road network after all threads terminate, and the results of route fitting are obtained, as shown in Fig. 6.

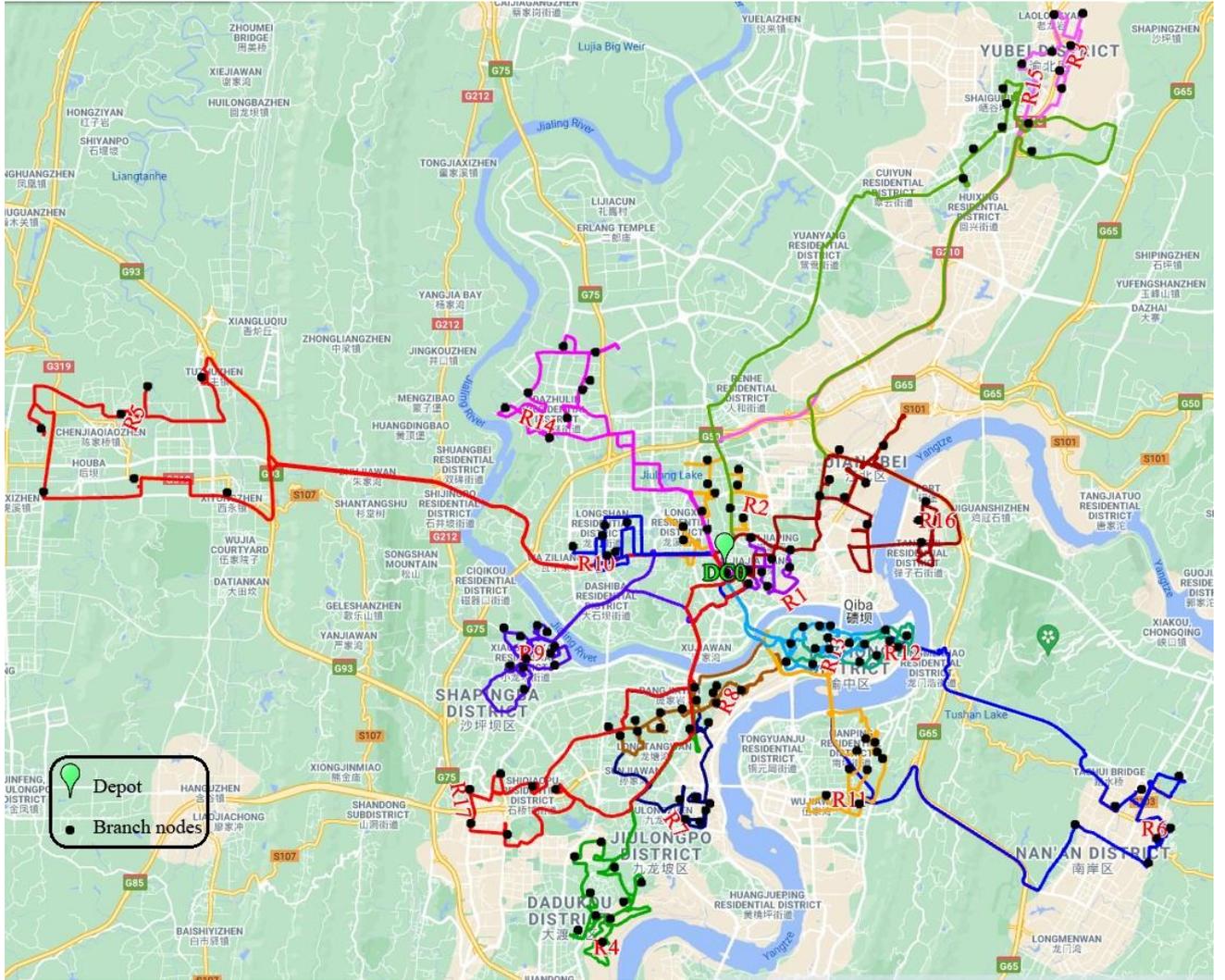


Fig. 6. Fitted routes based on Table 3.

Note that the NN-ILS algorithm designed in this paper is an online real-time optimization, and the real-time traffic information needs to be constantly obtained from Baidu Map during the running process. Therefore, different results may be obtained along with the changeable real-time traffic conditions, which means that the solution presented in Fig. 5 or Table 3 may vary depending on the initial starting time of the algorithm. Typically, the algorithm is used before CIT vehicles are about to depart from the depot, which ensures that the real-time traffic information obtained during the optimization process is up to date. Then, the route fitting procedure is used to guide the remote vehicles during the distribution process, and the detailed distribution routes are fitted into the real road network after completing the distribution process.

The impact of different departure times on the total distance and total emissions is studied due to the changeable traffic. The tour plan is simulated 13 times from 6:30 am to 18:30 pm on December 22, 2020, with an interval of 1 hour. The simulation results are presented in Table 4.

Table 4

The simulation results obtained at different departure times.

Departure time	ETA (min.)	Distance (km)	Emission (kg)
6:30	1419	665	172.0
7:30	2287	678	209.0
8:30	2307	688	209.4
9:30	1979	680	197.9
10:30	1804	668	188.9
11:30	1727	666	185.8
12:30	1627	667	181.6
13:30	1759	666	187.3
14:30	1902	669	193.1
15:30	1999	668	196.6
16:30	2301	668	206.9
17:30	3118	686	229.0
18:30	2914	682	225.0
Average	2088	673	199

It can be seen from Table 4 that different simulation results with different total ETA, distance, and emission can be obtained at different departure time based on the same tour plan. In fact, the company's vehicles depart at about 6.30 am for cash delivery and 15:30 pm for cash pick-up every day. Therefore, the morning delivery can be completed quickly with the lowest total emissions than others because it can avoid the rush hours. However, the pick-up process, however, hits the evening peak hours, leading to an increase of about 14 percent in total emissions and about 40 percent in total Estimated Time of Arrival (ETA) than the morning delivery process. According to Eq. (2), the total emissions are related not only to travel distance but also to the corresponding ETA because $v = d / t$. The total distances of the simulations did not fluctuate violently, so the correlation between total emissions and ETA is higher than that between total emissions and travel distance.

All experiments in this section are based on the use of DC0 as the depot, and the impact of dif-

ferent locations of the depot on costs will be discussed in the next section.

6.1.2 The impact of depot locations on costs

Although the DC0 is the best location as a depot because it ensures the shortest total distance between the depot and the clustering centers, the depot selected in the city center has many disadvantages, such as high rent, high parking costs and heavy traffic around. Therefore, the impacts of different locations on the total cost without considering the objective z_2 have been investigated in this section. The total cost is the summation of the vehicle fixed cost, distance cost and depot related cost. Five candidates are selected near the inner ring expressway (free access for local vehicles) in Chongqing. The housing rent, parking rent and total cost of different locations are presented in Table 5. The total area of the depot is 500 square meters and the number of parking is equal to the number of routes when calculating the total cost. In addition, the total travel distance is doubled, representing two trips per day for the same route.

Table 5

Costs comparison among different depots (the results obtained at 9:30~11:00am on December 23, 2020).

No.	Coordinates	Office and parking rent		Dis. (km)	Cost (Yuan RMB)
		Offi. (Yuan RMB.m ⁻²)	Park. (Yuan RMB.mon ⁻¹)		
DC0	11859739.76, 3429107.542	75	600	671.13	16201.5
DC1	11862290.67,3434710.742	50	400	808.9	15879.12
DC2	11859491.50,3433325.449	55	450	766.23	16741.53
DC3	11856293.32,3419135.634	55	450	849.81	17682.50
DC4	11864963.27,3421159.513	45	550	878.16	17621.83
DC5	11849584.20,3423822.816	45	500	939.47	18499.37

Therefore, the option by selecting DC1 as the depot has the lowest cost. Meanwhile, it is far away from the city center, so the traffic congestion is rare, which can increase the staff's satisfaction. Although the total distance of DC1 has increased, the total cost is the lowest due to the lower rent and parking fees than that in other locations.

6.1.3 Assessing security risk

It is necessary to evaluate the security risk of the CIT routes. The arc risk of transporting hazardous materials was defined as the product of the accident probability and the surrounding popula-

tion size [50]. Subsequently, they also used the Logit model to describe the accident probability of arcs more accurately [51]. Since the probability of accidents and the population size around arcs vary with time, the travel time and the risk of road segments were described as time-dependent functions $c_1^r(i, j)$ and $c_2^r(i, j)$, respectively [52]. Androutsopoulos and Zografos [53] added the on-board load as a parameter to calculate the risk index of routes, and suggested that factors such as wind speed and direction around the corresponding arc should be taken into account. Bula et al. [54] designed a complex exponential function to evaluate the distribution risks, in which a piece-wise linear function was employed to approximate the exponential function. Finally, a VNS algorithm was proposed to solve the routing problem. The VNS can also be applied for determining the optimal value of Support Vector Regression (SVR) parameters in a shorter time compared with the grid search or population-based metaheuristics [55]. The same route risk assessment method was adopted again [56], but different solution methods were adopted for the bi-objective vehicle routing problems. All the above-mentioned risk assessment methods are designed for hazardous materials transportation. Although these methods are helpful for assessing the CIT routes, they cannot be applied directly in the CTVRP model.

Talarico et al. [1] proposed a method for risk assessment by using the travel distance and the amount of money on board the vehicle:

$$R_j^r = R_i^r + d_{ij} M_i^r \quad (24)$$

where R_j^r is a cumulative risk index of the route r , M_i^r denotes the amount of money on board of the vehicle when it leaves node i along route r , and d_{ij} is the length of $arc(i, j)$ contained in r .

In this paper, the number of box pairs on board is used instead of the amount of money because the company does not know how much cash they transported. According to the data presented in Table 3, the risk index of the pick-up routes can be calculated (see Table 6).

Table 6

Cumulative risk along the routes during the pick-up process.

NO.	Risks of the nodes												
R01	0	2	6	9	17	22	34	41	65	74	104		
R02	0	1	5	11	31	41	65	72	96	105	115	126	186
R03	0	21	23	26	34	39	51	72	248				
R04	0	13	17	23	35	40	70	77	117	135	155	309	
R05	0	19	25	37	57	82	94	129	313				
R06	0	21	27	30	34	54	72	79	223				
R07	0	9	11	23	23	28	40	47	63	108	128	238	
R08	0	8	16	28	32	57	75	96	104	113	123	134	206
R09	0	8	10	16	24	44	62	76	100	109	129	140	272
R10	0	4	6	12	20	30	48	104					
R12	0	7	9	12	20	25	43	64	128				
R13	0	5	7	13	17	27	39	53	69	78	88	143	
R14	0	9	13	13	17	32	56	77	93	192			
R15	0	18	22	28	32	42	60	102	270				
R16	0	6	8	14	26	76	82	96	160	196	216	293	
R17	0	13	23	29	33	43	55	146					

As can be seen from Table 6, the risk in each node of the routes presents an increasing trend. In addition, the maximum value of the risk (313) is witnessed in R05 because the route (the red route on the left of Fig.6) serves the business district is far from the depot.

It is necessary for decision-makers of the CIT sectors to set a suitable risk threshold τ used for limiting the risk index of the routes. Consequently, it could also be a critical factor to be considered to negotiate fair insurance policies. If the decision-makers consider that the risk index of the route is too high, they can deal with it by adding more vehicles or enabling more depots.

The calculation method of the risk during the delivery process is different from Eq. (24) because the delivery risk is slowly released along the routes. Similarly, the delivery risk of the route r is defined as:

$$R_i^r = d_{i,i+1} M_i^r \quad (25)$$

where $d_{i,i+1}$ refers to the distance from node i to the next node, and we still use the number of box

pairs on broad when it leaves node i along route r to replace the cash amount M_i^r . Then, the risks of each route can be calculated during the delivery process (see Table 7).

Table 7

Risks along the routes during the delivery process.

NO.	Risks of the nodes												
R01	20	18	8	14	6	10	4	9	2	3	0		
R02	12	22	20	45	16	28	6	15	4	3	2	5	0
R03	168	7	6	10	4	6	6	22	0				
R04	143	20	18	24	7	30	5	20	6	4	14	0	
R05	152	21	24	25	20	6	10	23	0				
R06	168	21	6	5	16	9	2	18	0				
R07	99	10	36	0	7	12	5	8	15	4	10	0	
R08	96	44	40	9	40	21	18	5	4	3	2	6	0
R09	96	11	20	18	32	21	12	15	4	6	2	11	0
R10	28	6	10	8	6	6	8	0					
R11	100	9	0	7	18	15	12	6	2	9	0		
R12	56	7	6	10	4	9	6	8	0				
R13	55	10	18	8	14	12	10	8	3	2	5	0	
R14	81	16	0	6	15	16	9	4	11	0			
R15	144	14	12	5	8	9	12	21	0				
R16	66	10	18	24	70	6	10	32	12	4	7	0	
R17	91	30	10	4	6	4	13	0					

There are lower risks in morning delivery than afternoon pick-up. In Table 7, the risk value “0” in the middle of R07, R11 and R14 because of the errors caused by the rounding operation performed on distances (less than 0.5km) between two adjacent nodes in the calculation process.

6.2 Comparison with other algorithms

6.2.1 Introduction to algorithms compared

Three metaheuristic algorithms including ACO, SA, and TS are employed to test the performance of the NN-ILS algorithm. The parameters of the three algorithms used in this paper are presented in Table 8.

Table 8

Parameters of the metaheuristic algorithms.

Name	Parameters	Meaning	Value
ACO	<i>antNumber</i>	the number of ants	20
	<i>alpha</i>	pheromone decay parameter	0.1
	<i>beta</i>	the relative importance of pheromone	2
	<i>q₀</i>	random number uniformly distributed in [0,1]	0.9
	<i>maxIterations</i>	the number of iterations	100
	<i>initialTemperature</i>	initial temperature	100
	<i>lowestTemperature</i>	lowest temperature	0.001
SA	<i>updateRate</i>	temperature drop rate	0.99
	<i>stopNumber</i>	the maximum number of times that the optimal solution without change	50
	<i>maxIterations</i>	the number of iterations	2000
TS	<i>tabuListLength</i>	the length of tabu list	10
	<i>maxIterations</i>	the number of iterations	100

6.2.2 Comparison in results based on objective z_1

The greedy algorithm, NN algorithm and OR-Tools are also selected as the baselines to verify the accuracy of the proposed NN-ILS algorithm. All the seven algorithms are performed three times on the case of the CIT company and DC0 is selected as the depot. Finally, workdaythe best results obtained from 9:00 am to 12:00 am on December 23, 2020 are summarized in Table 9.

Table 9

The results calculated by different algorithms.

Algorithm	Vehicles	Distance (km)	Cost (Yuan RMB)
ACO	18	706.48	16984.81
SA	18	690.19	16960.25
TS	18	689.15	16958.69
Greedy	18	716.93	17000.56
OR-Tools	17	673.92	16205.9
NN	18	706.88	16985.41
NN-ILS	17	671.13	16201.53

According to Table 9, it is apparent that both the NN-ILS algorithm proposed in this paper and

OR-tools solver have achieved higher solution quality than other algorithms, while the NN-ILS algorithm is slightly better than OR-tools solver. However, due to the large number of parameters and random factors presented in the three metaheuristic algorithms, it is a difficult task to achieve an optimal parameter combination (hyper-parameters), which requires many times of experiments. Different results may be derived due to randomness, and the randomness makes the routes unpredictable, which may reduce the risk of potential robbery and increase the reliability of solutions. Meanwhile, it is worth pointing out that the results determined by the NN algorithm are fixed in the three times of executions because it selects the best result from the greedy algorithm at each starting point.

6.3 Uncertainty analysis and the application of MADM

6.3.1 The uncertainty of the algorithm

In order to analyze the uncertainty of the proposed NN-ILS algorithm, the real-time traffic information between any pairs of nodes is obtained from Baidu Map between 15:30 and 16:00 on January 8, 2021. This time interval coincides with the departure time for cash pick-up in the afternoon. Since the pick-up process is bound to hit the evening traffic peak, the effective working time T is set at 150 minutes. Then, 200 solutions, numbered 1-100 for minimizing z_1 and 101-200 for minimizing z_2 , are generated by the algorithm. Finally, the total distance and total emission of the solutions are illustrated in Fig. 7.

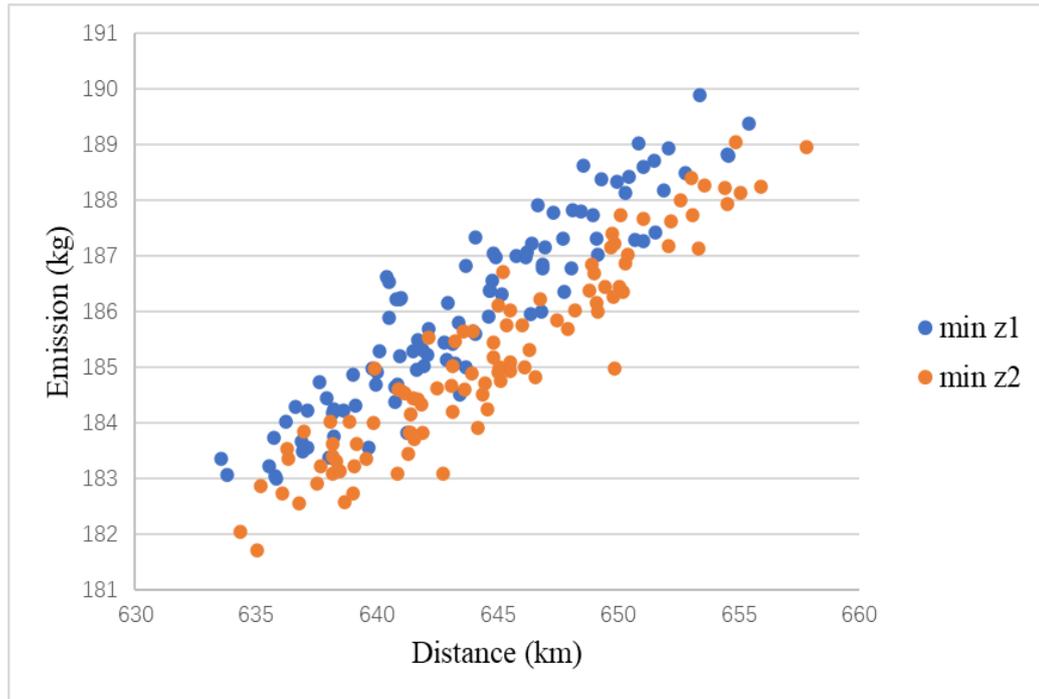


Fig. 7. The illustration based on the 200 solutions.

The results in Fig. 7 show that no matter which one is taken as the minimization objective, the output results of the algorithm are mainly concentrated in the middle shuttle region. There are only slight differences between the results obtained by minimizing different objectives: the solutions obtained by minimizing z_1 have a shorter total travel distance but a larger total emission, and vice versa. The two objectives show a strong correlation as the same real-time traffic information is employed, which makes it difficult to choose the best solution among the alternatives. Therefore, more attributes are evaluated for each solution, including the total travel distance, pollutant emissions, ETA, cost, CO₂ emission, congested distance, congested duration, and max risk (called global risk defined by Talarico et al. [1]). The purpose of splitting total emissions into CO₂ and pollutants is to give decision-makers a better understanding of the environmental impact of each solution. Then, the statistical characteristics of the solutions under the fixed real-time traffic information can be summarized in Table 10.

Table 10

The statistical characteristics of the 200 solutions with multiple attributes.

	Distance (km)	Pollutants (kg)	ETA (min.)	Cost (Yuan RMB)	CO ₂ (kg)	Congested dis- tance (km)	Congested dura- tion (min.)	Max risk
mean	644.27	1.465	2260	16161	184.14	8.97	66.66	315.89
std	5.38	0.014	25	8	1.75	2.74	18.48	1.55
min	633.54	1.430	2203	16144	180.28	3.18	28.00	315
25%	640.34	1.450	2243	16155	182.78	7.10	54.00	315
50%	643.96	1.460	2258	16160	183.99	8.39	63.00	315
75%	648.62	1.480	2278	16167	185.52	10.63	78.00	317
max	657.79	1.500	2326	16181	188.38	20.96	148.00	323

The data in Table 10 indicates that the designed algorithm is stable because each attribute of the solutions has small fluctuations. However, if real-time traffic information like Fig.8 is utilized in the optimization process, the range of feasible solutions may expand from a shuttle-shaped region to a similar-rectangular region (Fig. 9).

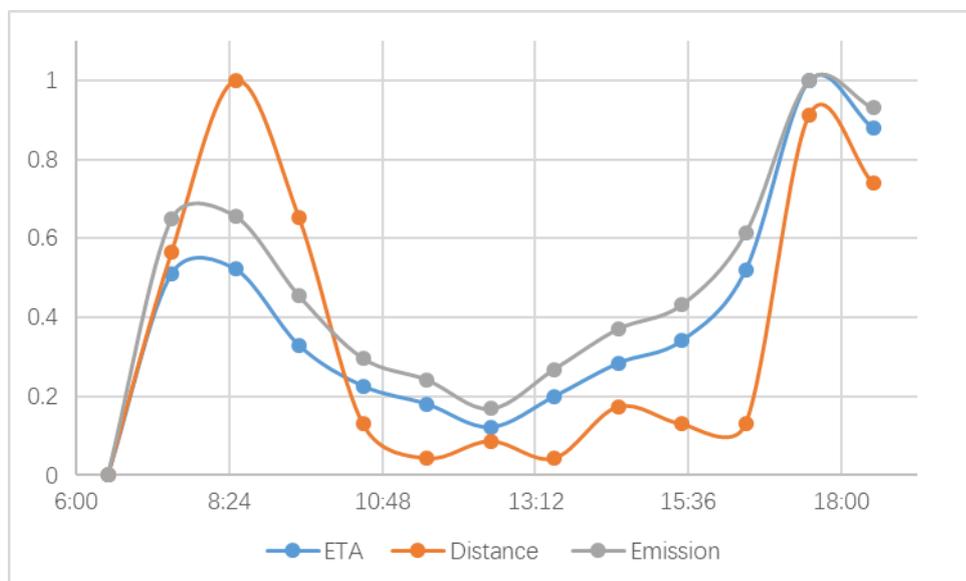


Fig. 8. The time-dependent fluctuations based on the normalized data of Table 4.

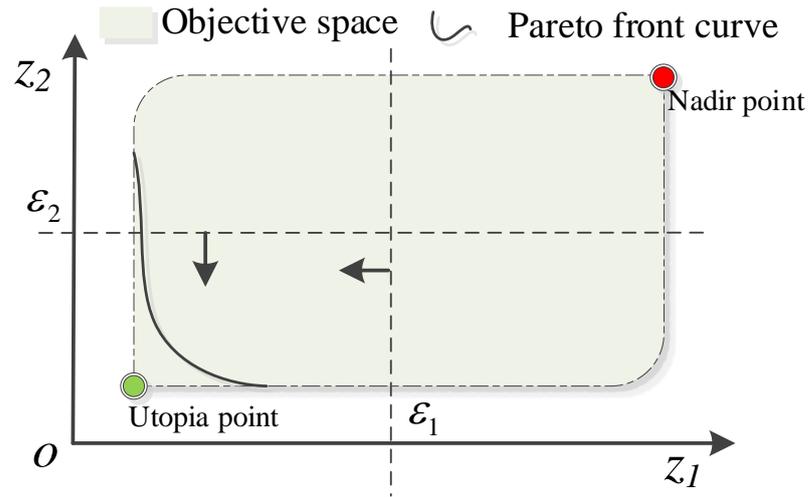


Fig. 9. Potential objective space and Pareto front curve.

At this point, if the ε -Constraint method [57] is adopted and appropriate decline rates of ε is set, many optimal solutions of Pareto front can be obtained, which may form a Pareto front curve. However, it is difficult to choose an optimal solution in the managerial sense on the objective space or Pareto frontier curve. The theory of MADM is utilized in the next section to select the most satisfactory solution from the objective space.

6.3.2 The application of MADM

The MCDM methods have been widely used in the optimization of supply chain. For example, Zandieh and Aslani [58] proposed a hybrid Genetic Algorithm integrated Analytical Hierarchy Process (GA-AHP) for order distribution in a multiple-supplier supply chain. Based on experts' opinion, the Best-Worst method and AHP were used to rank the solutions obtained by the GA-AHP sequentially. Similarly, the MADM is employed to identify the most satisfactory solution from these 200 solutions mentioned above. More specifically, PCA and Simple Additive Weighting (SAW) are combined to rank the feasible solutions.

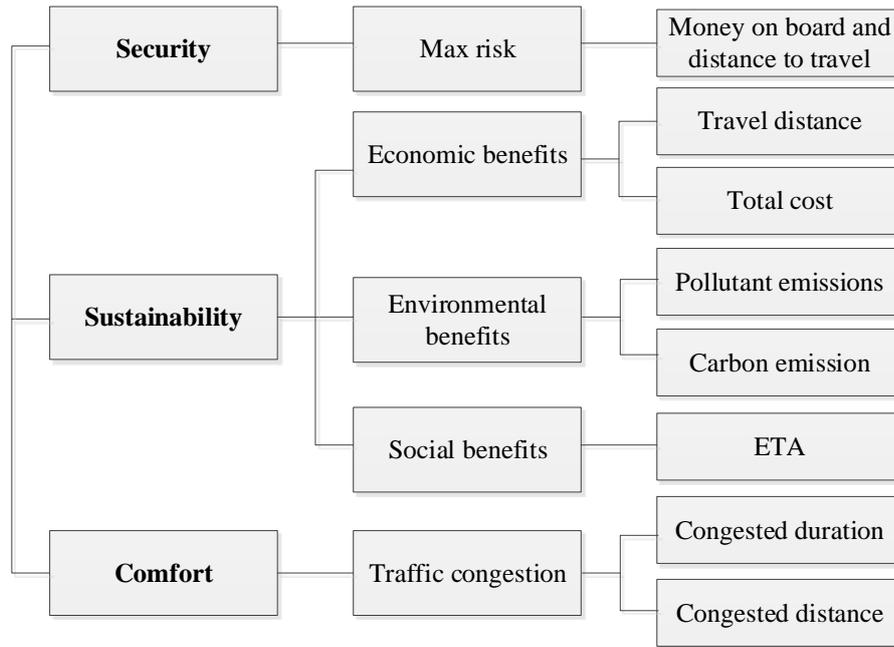


Fig. 10. Framework for evaluating performance of different solutions.

To achieve that, we have established a framework to evaluate the performance of these solutions, as presented in Fig. 10. The criteria in three dimensions are used for evaluating the performance of the solutions, and they are security, sustainability, and comfort. Among them, security mainly refers to the maximum risk of each solution, which is mainly measured by the index composed of travel distance and money on board, as presented in Eq. (24). The dimension of sustainability can be decomposed into three components: economic benefits, environmental benefits, and social benefits. The economic benefits are measured by two indicators: total travel distance and total cost, the economic benefits are represented by using the total pollutant emissions and the CO₂ emission, as described in Section 4, and the social benefits refer to total ETA, which measures the duration of all routes that satisfy all customers' demand. In terms of the dimension of comfort, it mainly considers the physical and mental benefits of drivers and escorts during work when experiencing traffic congestion. It can be measured by congested distance and corresponding congested duration, in which the severe congested speed is defined as below 10 km·h⁻¹ [59].

To verify the rationality and feasibility of this framework, we conduct the PCA for these eight indicators. The Kaiser-Meyer-Olkin (KMO) and Bartlett's Test show a KMO value of 0.719 with a

significant value of 0, which means that it can be analyzed by using PCA. The Screen Plot (see Fig. 11) shows that there are 3 eigenvalues greater than 1, indicating the 8 indicators can be represented by 3 components, and each of which explains 53.554%, 25.62%, and 13.728% of the total variance, or cumulatively 92.902%, as shown in Table 11. According to the component matrix presented in Table 12, CO₂ emission, pollutant emissions, travel distance, total cost, and ETA can be aggregated into the first component, congested duration and congested distance can be aggregated into the second component, and the last indicator, max risk, can be represented by the third components. This is consistent with the MADM framework developed in this study.

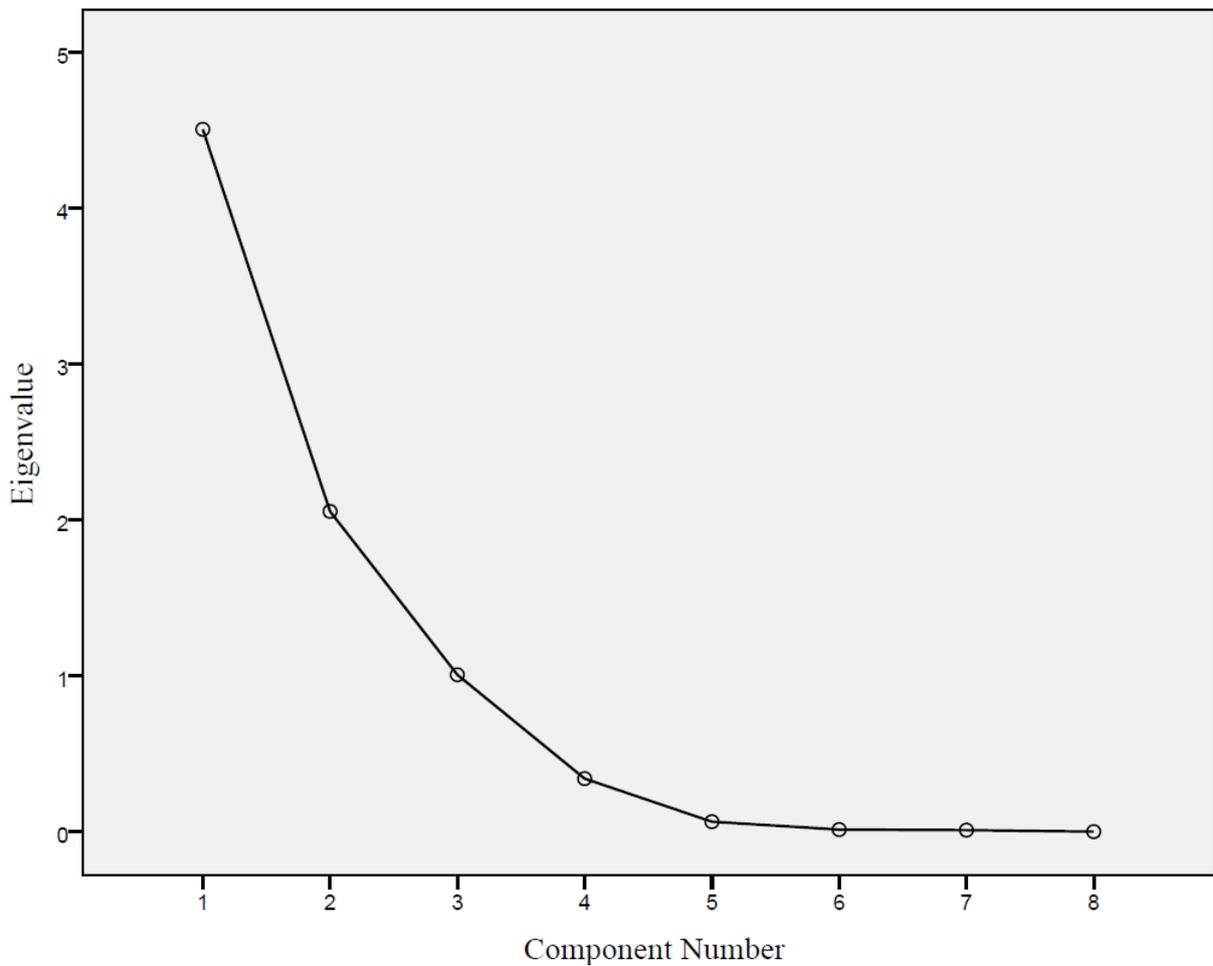


Fig. 11. Screen Plot in PCA.

Table 11

Total Variance Explained in PCA.

	Initial Eigenvalues			Extraction Sums of Squared Loadings			Rotation Sums of Squared Loadings		
	Total	% of		Total	% of		Total	% of	
		Variance	Cumulative %		Variance	Cumulative %		Variance	Cumulative %
1	4.284	53.554	53.554	4.284	53.554	53.554	4.281	53.513	53.513
2	2.050	25.620	79.174	2.050	25.620	79.174	2.050	25.623	79.136
3	1.098	13.728	92.902	1.098	13.728	92.902	1.101	13.766	92.902
4	.478	5.977	98.880						
5	.070	.877	99.756						
6	.011	.139	99.895						
7	.008	.097	99.992						
8	.001	.008	100.000						

Table 12

Component Matrix in PCA (3 components extracted).

	Component		
	1	2	3
Travel distance	.931	-.079	-.201
Pollutant emissions	.967	.007	.058
ETA	.795	.245	.301
Total cost	.931	-.082	-.203
CO ₂ emission	.990	-.004	.054
Congested distance	-.025	.992	-.038
Congested duration	-.021	.994	-.013
Max risk	.027	-.059	.958

With the information in Table 11 and Table 12, the weight with respect to each indicator, w_j , can be calculated, as shown in Table 13, and the procedure for the calculations can be referred to the work of Zeng et al. [60] and Rui et al. [61].

Table 13

The weights obtained by PCA.

Attribute	Weight
Travel distance	0.120066999
Pollutant emissions	0.155128644
ETA	0.173029797
Total cost	0.119604018
CO ₂ emission	0.157203641
Congested distance	0.099513964
Congested duration	0.102398293
Max risk	0.073054648

To determine the performance of each solution, their attribute values with respect to these eight indicators need to be normalized. As all these indicators are cost-type, their normalization should follow $a_{ij} = \frac{\max x_{ij} - x_{ij}}{\max x_{ij} - \min x_{ij}}$. Then the performance of each solution can be calculated by $P_i = w_j * a_{ij}$,

and the best satisfactory ten solutions are reported in Table 14 (the detailed information has been listed in the Appendix).

Table 14

The first ten solutions sorted by P in descending order.

NO.	Distance (km)	Pollutant emissions (kg)	ETA (min.)	Cost (Yuan RMB)	CO ₂ emission (kg)	Congested distance (km)	Congested duration (min.)	Max risk	P
188	634.37	1.43	2219	16146	180.61	7.33	55	315	0.91425
138	635.03	1.43	2203	16147	180.28	12.83	88	315	0.87772
114	638.69	1.44	2204	16152	181.13	6.32	48	315	0.87394
169	636.11	1.44	2228	16148	181.3	7.14	59	315	0.84860
181	639.01	1.44	2215	16153	181.3	8.05	60	315	0.83042
127	641.32	1.44	2206	16156	182	6.32	48	315	0.82829
179	637.55	1.45	2219	16150	181.45	7.27	56	315	0.82443
160	638.32	1.44	2222	16152	181.87	7.52	54	315	0.82425
81	633.54	1.45	2250	16144	181.9	6.25	51	315	0.82131
189	639.06	1.44	2218	16153	181.78	7.87	60	315	0.81765

The data presented in Table 14 shows that the MADM approach pays more attention to the

emissions rather than the distances and others. To make a more intuitive comparison between the best and worst solutions, the routes of the two solutions are displayed in Fig. 12.

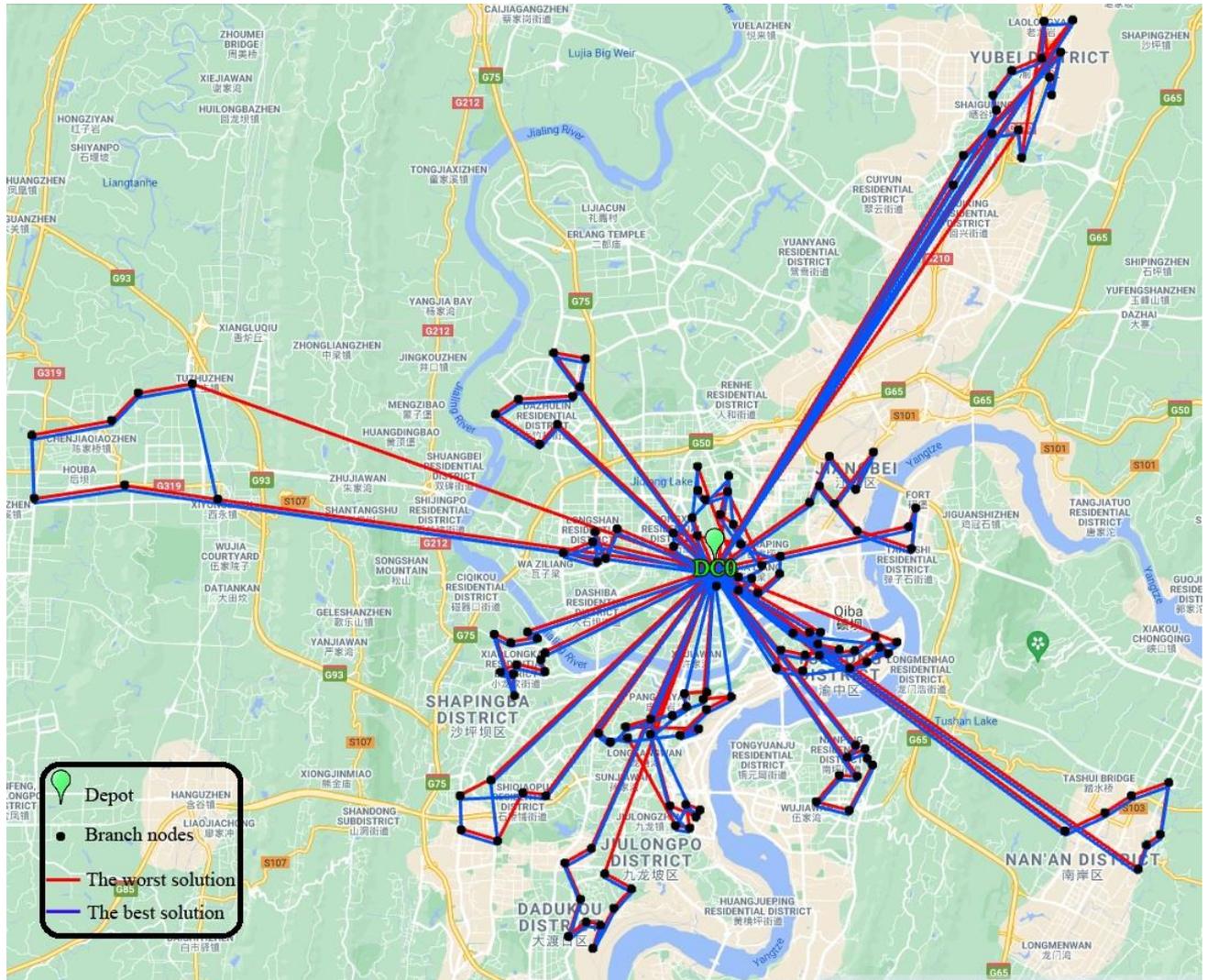


Fig. 12. The comparison between the best and worst solutions.

According to Fig. 12, the red lines represent the worst solution and the blue lines denote the best solution. The blue lines have been shifted slightly down for convenience. It is apparent that there are seven absolutely overlapping routes, while the other non-overlapping paths are mainly distributed in areas with dense customer nodes.

Briefly, the experimental results of the case study show that: (1) the model and solution framework designed in this paper are suitable for the CTVRP under the scenario with clustered customers.

The depot insertion strategy in the splitting process provides a simple and flexible method for selecting the best depot location; (2) the solution quality of NN-ILS algorithm is better than that of the classical algorithms such as ACO, SA and TS, as well as slightly better than OR-Tools; (3) the uncertainty of NN-ILS algorithm is analyzed by using the repeated experiment method, and the most satisfactory solution can be determined objectively from multiple alternative solutions based on MADM, so as to eliminate the influence of decision-makers' subjective judgments in the selection process.

7 Conclusions

To improve the effectiveness of the cash distribution, this study presented a CTVRP model and proposed a novel solution framework considering the special terrain by using the real-time traffic information. Meanwhile, the route fitting procedure can not only guide remote vehicles to avoid congestion in real time, but also fit the detailed distribution routes into the actual traffic network, which are beneficial to both the decision-makers and the drivers of CIT sectors. Finally, the feasibility and effectiveness of the algorithm are verified by comparing it with other well-known algorithms.

The model and algorithm proposed in this paper have the following features:(1) The strategy determined by the proposed model can avoid congestion when obtaining the real-time traffic information during the optimization and distribution process. The strategy can reduce the waiting time caused by congestion as studies have shown that vehicles cause much more environmental pollution in idling than the normal operations [62]. Therefore, the proposed solution framework may diminish the negative impacts of the cash distribution process on the environment. (2) The evaluation framework on solutions is helpful for decision-makers to objectively choose the most satisfactory solution among multiple alternatives. (3) This is a generic model, and this paradigm can also be used for the transportation of other valuable goods (e.g., antiques, gold, silver, and jewelry, etc.).

However, there are still some weaknesses in this study, and more work will be done in future to

overcome these weaknesses. First, the proposed model only considers the economic and the environmental objectives. Multi-objective models capturing conflicting or partial conflicting objectives could be developed such as the number of vehicles used, the satisfaction of customers or CIT drivers. Second, although the *K-means* algorithm has advantages in effect of clustering and the low complexity, it is not suitable for the classification of the samples with high discrete or non-convex distribution. More sophisticated spectral clustering algorithms [63] could adapt to the classification of diversified sample distributions. Third, many factors (e.g., the economic and social development, the level of crime in target areas) are overlooked in the risk assessment method used in this paper. Therefore, comprehensive risk assessment methods that consider additional factors should be developed in the future research.

Acknowledgments

This research was funded by the Chongqing graduate Scientific research innovation Project [grant number CYB20178, 2020], and National Social Science Foundation of China [grant number 19CGL041, 2019].

Appendix A. Full list of abbreviations that were used in the manuscript

CIT	Cash-in-Transit
CTVRP	Cash-in-Transit Vehicle Routing Problem
NN-ILS	Nearest Neighbor-first Iterated Local Search-second algorithm
MADM	Multi-Attribute Decision Making
PCA	Principal Component Analysis
VRP	Vehicle Routing Problem
IEA	International Energy Agency
VRPTW	Vehicle Routing Problem with Time Windows
ACO	Ant Colony Optimization
GA	Genetic Algorithm
VNS	Variable Neighborhood Search
TS	Tabu Search
SA	Simulated Annealing
RCTVRP	Risk-Constrained Cash-in-Transit Vehicle Routing Problem
LNS	Large Neighborhood Search
SAEA	Self-Adaptive parallel Evolutionary Algorithm
NN	Nearest Neighbor
VRPBTW	Vehicle Routing Problem with Backhauls and Time Windows
GVRP	Green Vehicle Routing Problem
ILS	Iterated Local Search
TSP	Travelling Salesman Problem
COPERT	Computer Programme to calculate Emissions from Road Transport
MEET	Methodologies for Estimating air pollutant Emissions from Transport
NAEI	National Emissions Inventory
CMEM	Comprehensive Modal Emission Model

HDV	Heavy-Duty Vehicle
LDV	Light-Duty Vehicle
DC	Distribution Center
QoS	Quality of Service
ETA	Estimated Time of Arrival
SVR	Support Vector Regression
GA-AHP	Genetic Algorithm integrated Analytical Hierarchy Process
SAW	Simple Additive Weighting
KMO	Kaiser-Meyer-Olkin

Appendix B. The solutions sorted by P in descending order

NO.	Distance (km)	Pollutant emissions (kg)	ETA (min.)	Cost (Yuan RMB)	CO2 (kg)	Congested distance (km)	Congested duration (min.)	Max risk	P
188	634.37	1.43	2219	16146	180.61	7.33	55	315	0.91425
138	635.03	1.43	2203	16147	180.28	12.83	88	315	0.87772
114	638.69	1.44	2204	16152	181.13	6.32	48	315	0.87394
169	636.11	1.44	2228	16148	181.3	7.14	59	315	0.84860
181	639.01	1.44	2215	16153	181.3	8.05	60	315	0.83042
127	641.32	1.44	2206	16156	182	6.32	48	315	0.82829
179	637.55	1.45	2219	16150	181.45	7.27	56	315	0.82443
160	638.32	1.44	2222	16152	181.87	7.52	54	315	0.82425
81	633.54	1.45	2250	16144	181.9	6.25	51	315	0.82131
189	639.06	1.44	2218	16153	181.78	7.87	60	315	0.81765
113	636.32	1.44	2242	16149	181.91	7.08	54	315	0.81740
144	638.18	1.45	2227	16151	181.64	6.32	48	315	0.81528
4	635.86	1.45	2225	16148	181.54	7.43	59	317	0.80735
177	636.76	1.44	2219	16149	181.11	7.04	48	322	0.80452
142	639.16	1.45	2227	16153	182.17	6.32	48	315	0.79368
84	635.52	1.45	2240	16147	181.77	7.34	55	317	0.79062
128	640.84	1.44	2226	16155	181.64	8.53	65	315	0.78587
115	637.66	1.45	2228	16151	181.76	9.55	71	315	0.77641
145	638.45	1.44	2231	16152	181.7	10.86	82	315	0.77166
91	638.04	1.44	2236	16151	181.93	11.15	75	315	0.76977
25	635.78	1.45	2238	16148	181.6	7.09	54	320	0.76707
98	636.91	1.45	2244	16149	182.04	7.33	56	317	0.76561

29	637.14	1.45	2235	16150	182.11	8.35	62	317	0.76171
103	642.76	1.45	2207	16158	181.63	9.55	71	315	0.76060
176	638.17	1.45	2236	16151	182.18	9.07	67	315	0.76058
108	641.34	1.45	2229	16156	182.38	7.31	55	315	0.75478
75	633.78	1.44	2254	16145	181.63	12.33	93	317	0.75053
12	641.25	1.45	2227	16156	182.36	8.06	60	315	0.74996
120	638.06	1.45	2248	16151	182.57	7.87	60	315	0.74937
119	638.16	1.45	2239	16151	181.94	9.96	75	315	0.74926
136	641.39	1.45	2237	16156	182.71	6.37	49	315	0.74726
151	635.2	1.45	2239	16147	181.42	14.23	99	315	0.74256
19	636.26	1.45	2250	16148	182.56	10.08	74	315	0.74104
186	641.85	1.46	2231	16157	182.87	5.12	39	315	0.74045
193	641.38	1.45	2224	16156	182.36	9.55	71	315	0.73581
143	641.88	1.45	2224	16157	182.37	9.65	64	315	0.73533
122	641.55	1.45	2223	16156	182.26	10.16	74	315	0.73235
104	643.12	1.45	2225	16159	182.75	7.94	58	315	0.72863
34	638.24	1.45	2239	16151	182.3	11.31	84	315	0.72664
110	636.96	1.45	2257	16150	182.39	10.19	76	315	0.72224
87	640.85	1.45	2248	16155	183.23	7.33	55	315	0.71710
187	636.28	1.45	2251	16148	182.07	13.19	93	315	0.71543
7	637.65	1.46	2261	16151	183.26	4.87	39	317	0.71400
60	638.6	1.45	2253	16152	182.76	7.73	62	317	0.71355
132	643.08	1.45	2235	16159	183.22	8.07	55	315	0.70747
121	645.55	1.46	2225	16162	183.48	4.62	37	315	0.70707
33	636.88	1.45	2253	16149	182.21	11.26	85	317	0.70306
92	639.98	1.46	2256	16154	183.22	6.32	48	315	0.70305
150	644.17	1.45	2222	16160	182.46	10.83	72	315	0.70193
32	639.66	1.45	2243	16154	182.11	10.63	79	317	0.69778
58	636.63	1.46	2264	16149	182.83	8.01	60	317	0.69415
13	635.75	1.45	2262	16148	182.28	11.67	90	317	0.69130
185	638.85	1.46	2252	16152	182.56	9.91	75	315	0.69041
164	641.68	1.46	2241	16157	182.95	8.42	65	315	0.68501
195	643.62	1.46	2244	16160	183.13	6.32	48	315	0.68426
156	646.58	1.46	2224	16164	183.37	6.44	44	315	0.68289
158	639.87	1.45	2242	16154	182.55	12.97	94	315	0.68198
101	644.38	1.46	2227	16161	183.05	8.53	63	315	0.67756
24	639.13	1.45	2254	16153	182.86	10.34	77	317	0.67694
152	645.52	1.46	2233	16162	183.63	6.44	45	315	0.67604
126	640.93	1.45	2258	16156	183.13	9.46	71	315	0.67577
46	637.15	1.45	2263	16150	182.78	11.24	83	317	0.67518
36	639	1.46	2265	16153	183.4	6.43	51	317	0.67354
51	640.78	1.46	2248	16155	183.18	7.94	58	317	0.67202
77	641.52	1.46	2256	16156	183.82	4.92	43	317	0.67115
15	641.06	1.46	2239	16156	183.1	11.04	75	315	0.66802

73	642.12	1.46	2249	16157	183.75	5.89	47	317	0.66731
107	644.59	1.45	2225	16161	182.79	12.16	86	315	0.66660
171	645.02	1.46	2234	16162	183.45	7.19	57	315	0.66617
21	638.2	1.46	2254	16151	182.78	6.81	52	322	0.66283
9	637.94	1.46	2267	16151	182.99	8.94	69	317	0.66099
111	643.95	1.46	2243	16160	183.43	7.85	61	315	0.65855
200	639.58	1.45	2245	16153	181.91	15.87	119	315	0.65728
154	649.83	1.46	2216	16169	183.52	6.32	48	315	0.65624
90	642.88	1.46	2248	16158	183.67	6.77	52	317	0.65408
106	645.14	1.46	2237	16162	183.3	8.33	65	315	0.65106
161	646.31	1.46	2252	16164	183.85	4.96	40	315	0.64722
93	638.17	1.45	2264	16151	182.72	13.21	93	317	0.64709
2	639.99	1.45	2278	16154	183.45	8.2	66	317	0.64560
117	641.48	1.46	2253	16156	182.98	10.8	84	315	0.64224
182	644.5	1.46	2243	16161	183.25	9.21	69	315	0.64165
96	643.69	1.46	2242	16160	183.54	9.91	75	315	0.63563
71	639.8	1.46	2258	16154	183.52	11.68	83	315	0.63544
37	640.77	1.46	2247	16155	182.91	13.31	97	315	0.63365
16	641.68	1.47	2257	16157	184.02	7.87	60	315	0.62692
38	640.11	1.46	2271	16154	183.83	10.28	75	315	0.62426
8	643.4	1.46	2265	16159	184.33	5.65	45	317	0.62379
172	647.46	1.46	2245	16165	184.38	6.32	48	315	0.62342
148	645.08	1.46	2243	16162	183.54	9.67	75	315	0.62222
94	643.25	1.46	2257	16159	183.61	8.35	62	317	0.62015
80	644.07	1.47	2255	16160	184.12	5.27	41	317	0.61877
199	641.16	1.46	2258	16156	183.07	12.98	89	315	0.61858
192	645.03	1.46	2274	16162	184.64	4.72	40	315	0.61508
22	640.97	1.46	2255	16156	183.73	10.6	79	317	0.61452
180	643.61	1.46	2276	16160	184.19	7.33	55	315	0.60709
159	646.13	1.46	2243	16163	183.54	10.63	79	315	0.60501
116	643.26	1.47	2272	16159	183.99	6.7	54	315	0.60378
3	641.65	1.46	2249	16157	183.49	9.7	74	320	0.60293
129	642.5	1.45	2257	16158	183.16	14.8	107	315	0.60175
6	640.95	1.46	2286	16156	184.77	6.22	52	317	0.59838
191	643.15	1.46	2257	16159	183.55	12.38	87	315	0.59618
183	645.36	1.47	2260	16162	184.29	7.1	50	315	0.59592
125	645.24	1.47	2274	16162	185.23	4.28	31	315	0.59058
64	643.15	1.47	2257	16159	183.95	8.39	65	317	0.58910
190	644.83	1.46	2254	16161	183.72	11.78	87	315	0.58568
52	640.53	1.47	2295	16155	184.41	7.33	55	315	0.58535
141	646.04	1.47	2253	16163	184.29	8.53	63	315	0.58007
97	642.15	1.46	2279	16157	184.23	9.08	69	317	0.57902
109	648.21	1.46	2249	16166	184.55	8.69	67	315	0.57807
50	646.35	1.46	2261	16164	184.49	8.89	68	315	0.57605

10	647.75	1.47	2248	16166	184.89	5.46	40	317	0.57584
11	640.51	1.47	2290	16155	185.06	7.85	59	315	0.57354
153	647.93	1.47	2243	16166	184.21	8.89	67	315	0.57121
118	649.1	1.47	2242	16168	184.69	7.33	55	315	0.57001
175	650.19	1.47	2242	16169	184.89	6.32	48	315	0.56913
95	645.17	1.47	2268	16162	184.84	7.3	60	315	0.56528
23	644.72	1.47	2266	16161	184.89	7	50	317	0.56453
49	640.79	1.47	2290	16155	184.76	7.65	56	317	0.56339
88	642.78	1.46	2267	16158	183.98	8.01	63	322	0.55985
162	645.52	1.47	2273	16162	184.56	7.91	60	315	0.55854
26	641.95	1.46	2272	16157	183.57	14.81	107	315	0.55643
83	643.97	1.47	2266	16160	184.17	11.21	78	315	0.55626
197	643.97	1.47	2266	16160	184.17	11.21	78	315	0.55626
196	639.93	1.46	2277	16154	183.52	16.38	113	315	0.55616
146	649.17	1.47	2246	16168	184.53	8.53	63	315	0.55360
131	646.75	1.47	2265	16164	184.76	8.06	60	315	0.55251
31	640.42	1.47	2301	16155	185.14	8.42	63	315	0.55036
69	641.89	1.46	2258	16157	183.85	10.61	80	323	0.54448
41	642.94	1.46	2291	16159	184.7	8.82	69	317	0.54409
55	641.01	1.47	2290	16156	184.78	8.85	66	317	0.54343
139	649.46	1.47	2258	16168	184.98	6.91	54	315	0.54330
66	644.77	1.47	2272	16161	185.09	9.33	67	315	0.54268
198	649.78	1.47	2253	16169	184.79	7.87	60	315	0.53871
63	643.42	1.46	2255	16159	183.05	16.48	120	317	0.53799
163	642.14	1.47	2284	16157	184.07	12.98	89	315	0.53235
147	650.04	1.47	2251	16169	184.98	8.48	61	315	0.53228
174	644.85	1.46	2262	16161	183.99	15.75	108	315	0.52894
59	644.65	1.47	2270	16161	184.44	10.63	79	317	0.52292
167	648.83	1.47	2260	16167	184.9	9.85	75	315	0.51402
14	646.8	1.47	2262	16164	184.53	6.88	54	322	0.50875
68	646.87	1.47	2275	16164	185.38	7.91	60	317	0.50840
72	644.91	1.48	2277	16162	185.5	9.05	67	315	0.50317
184	649.01	1.48	2264	16168	185.21	7.99	56	315	0.50271
65	644.08	1.48	2294	16160	185.86	7.27	56	315	0.50219
135	648.93	1.48	2267	16168	185.36	7.58	58	315	0.49656
67	644.82	1.48	2288	16161	185.56	7.31	55	317	0.49193
137	650.31	1.48	2262	16170	185.39	7.94	58	315	0.48770
79	646.89	1.47	2281	16164	185.3	9.21	69	317	0.48645
140	649.87	1.48	2273	16169	185.73	6.84	51	315	0.48317
105	650.39	1.48	2274	16170	185.53	7.94	58	315	0.46771
82	649.15	1.47	2279	16168	185.54	10.25	78	315	0.46525
102	649.74	1.48	2271	16169	185.91	8.51	63	315	0.46355
42	643.69	1.48	2296	16160	185.33	11.61	87	315	0.46085
86	646.19	1.48	2284	16163	185.5	9.39	71	317	0.46017

62	648.94	1.48	2289	16168	186.26	5.64	42	317	0.45435
166	650.12	1.48	2287	16169	186.24	6.62	51	315	0.45357
134	652.21	1.48	2270	16173	186.13	6.7	54	315	0.45334
61	649.96	1.49	2280	16169	186.85	3.18	28	317	0.45083
168	653.34	1.48	2254	16174	185.65	9.46	71	315	0.44638
170	653.08	1.48	2270	16174	186.25	6.72	51	315	0.44591
100	648.04	1.47	2277	16166	185.31	12.45	88	317	0.44538
149	654.53	1.48	2262	16176	186.46	6.18	50	315	0.44332
99	650.28	1.48	2295	16170	186.64	6.16	48	315	0.43566
44	646.2	1.48	2294	16163	185.58	10.54	74	317	0.43551
43	645.77	1.48	2285	16163	185.51	12	84	317	0.43495
1	647.73	1.48	2281	16166	185.83	10.07	76	317	0.43259
130	652.08	1.48	2264	16172	185.7	10.89	78	315	0.43007
40	646.4	1.48	2282	16164	185.73	11.74	86	317	0.42830
27	647.33	1.48	2299	16165	186.29	9.67	73	315	0.42662
157	654.44	1.49	2272	16176	186.72	4.2	35	315	0.42638
178	651.06	1.48	2290	16171	186.19	7.87	60	315	0.42453
17	650.44	1.48	2292	16170	186.95	5.54	44	317	0.42170
89	646.69	1.48	2304	16164	186.43	5.89	48	320	0.42010
54	649.31	1.49	2295	16168	186.88	4.87	39	317	0.41675
165	652.59	1.48	2282	16173	186.52	7.94	58	315	0.41665
76	650.68	1.48	2270	16170	185.81	10.78	80	317	0.41353
123	649.72	1.48	2277	16169	185.67	13.04	91	315	0.41062
30	651.02	1.49	2291	16171	187.1	5.67	45	315	0.40861
85	646.96	1.48	2300	16165	185.67	12.41	92	315	0.40753
56	649.13	1.48	2279	16168	185.83	7.91	58	322	0.40380
173	655.07	1.48	2274	16177	186.66	7.09	54	316	0.39902
133	653.57	1.48	2288	16175	186.79	7.33	55	315	0.39763
47	651.89	1.48	2285	16172	186.7	8.36	64	317	0.38990
112	653.03	1.49	2284	16174	186.9	7.92	59	315	0.37815
45	651.56	1.48	2278	16172	185.93	9.72	77	320	0.37023
78	648.13	1.48	2293	16166	186.34	14	98	317	0.36307
53	648.48	1.49	2312	16167	186.31	11.6	88	315	0.35002
155	655.89	1.49	2265	16178	186.76	11.21	78	315	0.34588
20	652.8	1.48	2295	16173	187.01	11.73	87	315	0.34185
57	654.52	1.49	2291	16176	187.33	8.38	64	315	0.33927
70	651.02	1.47	2287	16171	185.8	17.55	125	317	0.33077
39	650.86	1.49	2305	16170	187.52	10.93	81	315	0.32463
18	655.43	1.49	2299	16177	187.88	6.39	50	317	0.31443
194	657.79	1.49	2284	16181	187.47	8.53	63	315	0.31406
5	654.56	1.49	2289	16176	187.31	10.79	79	317	0.29772
124	654.88	1.49	2307	16177	187.56	9.69	70	315	0.29483
74	648.55	1.49	2321	16167	187.14	13.56	90	317	0.28997
35	652.09	1.49	2316	16172	187.44	11.19	86	317	0.27417

48	653.36	1.5	2323	16174	188.38	12.02	91	317	0.20225
28	651.47	1.49	2326	16171	187.23	20.96	148	315	0.18115

References

1. Talarico, L., K. Sörensen, and J. Springael, *Metaheuristics for the risk-constrained cash-in-transit vehicle routing problem*. European Journal of operational research, 2015. **244**(2): p. 457-470.
2. Radojicic, N., A. Djenic, and M. Maric, *Fuzzy GRASP with path relinking for the Risk-constrained Cash-in-Transit Vehicle Routing Problem*. Applied Soft Computing, 2018. **72**: p. 486-497.
3. Tikani, H., M. Setak, and E. Demir, *Multi-objective periodic cash transportation problem with path dissimilarity and arrival time variation*. Expert Systems with Applications, 2021. **164**: p. 114015.
4. Fernández, R.Á., S.C. Caraballo, and F.C. López, *A probabilistic approach for determining the influence of urban traffic management policies on energy consumption and greenhouse gas emissions from a battery electric vehicle*. Journal of Cleaner Production, 2019. **236**.
5. Golpîra, H., S.A.R. Khan, and S. Safaeipour, *A review of logistics Internet-of-Things: Current trends and scope for future research*. Journal of Industrial Information Integration, 2021. **22**: p. 100194.
6. Solomon, M.M., *Algorithms for the Vehicle Routing and Scheduling Problems with Time Window Constraints*. Operations Research, 1987. **35**(2): p. 13.
7. El-Sherbeny, N.A., *Vehicle routing with time windows: An overview of exact, heuristic and metaheuristic methods*. Journal of King Saud University - Science, 2010. **22**(3): p. 123-131.
8. Prins, C., *Efficient Heuristics for the Heterogeneous Fleet Multitrip VRP with Application to a Large-Scale Real Case*. Journal of Mathematical Modelling and Algorithms, 2002. **1**(2): p. 135-150.
9. Clarke, G. and J.W. Wright, *Scheduling of Vehicles from a Central Depot to a Number of Delivery Points*. Operations Research, 1964. **16**(4): p. 15.
10. Pureza, V., R. Morabito, and M. Reimann, *Vehicle routing with multiple deliverymen: Modeling and heuristic approaches for the VRPTW*. European Journal of Operational Research, 2012. **218**(3): p. 636-647.
11. Zhang, H., et al., *A hybrid ant colony optimization algorithm for a multi-objective vehicle routing problem with flexible time windows*. Information Sciences, 2019. **490**: p. 166-190.
12. Salavati-Khoshghalb, M., et al., *An exact algorithm to solve the vehicle routing problem with stochastic demands under an optimal restocking policy*. European Journal of Operational Research, 2019. **273**(1): p. 175-189.
13. Pasha, J., et al., *An Optimization Model and Solution Algorithms for the Vehicle Routing Problem With a "Factory-in-a-Box"*. IEEE Access, 2020. **8**: p. 134743-134763.
14. Mojtahedi, M., et al., *Sustainable vehicle routing problem for coordinated solid waste management*. Journal of Industrial Information Integration, 2021. **23**: p. 100220.
15. Tarantilis, C.D. and C.T. Kiranoudis, *An adaptive memory programming method for risk logistics operations*. International Journal of Systems Science, 2004. **35**(10): p. 12.

16. Talarico, L., et al., *A large neighbourhood metaheuristic for the risk-constrained cash-in-transit vehicle routing problem*. Computers & Operations Research, 2017. **78**: p. 547-556.
17. Chang, Y.H., *The Cash Pick-up and Delivery Vehicle Routing/Scheduling under Stochastic Travel Times*. 2011, National Central University: Taiwan.
18. Yan, S.Y., S.S. Wang, and M.W. Wu, *A model with a solution algorithm for the cash transportation vehicle routing and scheduling problem*. Computers & Industrial Engineering, 2012. **63**(2): p. 464-473.
19. Boonsam, P., N. Suthikarnnarunai, and W. Chitphaiboon, *Assignment Problem and Vehicle Routing Problem for an Improvement of Cash Distribution*. Proceedings of the World Congress on Engineering and Computer Science, 2011: p. 5.
20. Sabar, N.R., et al., *A self-adaptive evolutionary algorithm for dynamic vehicle routing problems with traffic congestion*. Swarm and Evolutionary Computation, 2019. **44**: p. 1018-1027.
21. Theophilus, O., et al., *Truck scheduling optimization at a cold-chain cross-docking terminal with product perishability considerations*. Computers & Industrial Engineering, 2021. **156**: p. 107240.
22. Küçüköglu, İ. and N. Öztürk, *An advanced hybrid meta-heuristic algorithm for the vehicle routing problem with backhauls and time windows*. Computers & Industrial Engineering, 2015. **86**: p. 60-68.
23. Mohammed, M.A., et al., *Solving vehicle routing problem by using improved K-nearest neighbor algorithm for best solution*. Journal of Computational Science, 2017. **21**: p. 232-240.
24. Talavera-Llames, R., et al., *MV-kWNN: A novel multivariate and multi-output weighted nearest neighbours algorithm for big data time series forecasting*. Neurocomputing, 2019. **353**: p. 56-73.
25. Bräysy, O. and M. Gendreau, *Vehicle Routing Problem with Time Windows, Part I: Route Construction and Local Search Algorithms*. Transportation Science, 2005. **39**(1): p. 104-118.
26. Braysy, O., G. Hasle, and W. Dullaert, *A multi-start local search algorithm for the vehicle routing problem with time windows*. European Journal of Operational Research, 2004. **159**(3): p. 586-605.
27. Andelmin, J. and E. Bartolini, *A multi-start local search heuristic for the Green Vehicle Routing Problem based on a multigraph reformulation*. Computers & Operations Research, 2019. **109**: p. 43-63.
28. Toffolo, T.A.M., et al., *Stochastic local search with learning automaton for the swap-body vehicle routing problem*. Computers & Operations Research, 2018. **89**: p. 68-81.
29. Lourenco, H.R., O.C. Martin, and T. Stützle, *Iterated Local Search Framework and Applications*. International Series in Operations Research & Management Science, 2010.
30. Hosseinabadi, A.A.R., et al., *A new efficient approach for solving the capacitated Vehicle Routing Problem using the Gravitational Emulation Local Search Algorithm*. Applied Mathematical Modelling, 2017. **49**: p. 663-679.
31. Brandão, J., *Iterated local search algorithm with ejection chains for the open vehicle routing problem with time windows*. Computers & Industrial Engineering, 2018. **120**: p. 146-159.
32. Bell, J.E. and P.R. McMullen, *Ant colony optimization techniques for the vehicle routing problem*. Advanced Engineering Informatics, 2004. **18**(1): p. 41-48.
33. Maity, S., A. Roy, and M. Maiti, *An intelligent hybrid algorithm for 4- dimensional TSP*. Journal of Industrial Information Integration, 2017. **5**: p. 39-50.
34. Breedam, A.V., *Improvement heuristics for the Vehicle Routing Problem based on*

- simulated annealing*. European Journal of Operational Research, 1995. **86**(1995): p. 11.
35. Gendreau, M., A. Hertz, and G. Laporte, *A Tabu Search Heuristic for the Vehicle Routing Problem*. Management Science, 1994. **40**(10): p. 16.
 36. Google. *Solving a TSP with OR-Tools*. 2021 [cited 2021 7 January]; Available from: <https://developers.google.cn/optimization/routing/tsp>.
 37. Bello, I., et al., *Neural combinatorial optimization with Reinforcement Learning*. ICLR Workshop, 2017.
 38. Nazari, M., et al., *Deep Reinforcement Learning for Solving the Vehicle Routing Problem*. International Conference on Machine Learning, 2018.
 39. Kool, W., H.v. Hoof, and M. Welling, *Attention, Learn to Solve Routing Problems!* ICLR Workshop, 2019.
 40. Prins, C., P. Lacomme, and C. Prodhon, *Order-first split-second methods for vehicle routing problems: A review*. Transportation Research Part C: Emerging Technologies, 2014. **40**: p. 179-200.
 41. Ntziachristos, L. and Z. Samaras, *COPERT III, Computer programme to calculate emissions from road transport*. 2000, Copenhagen: European Energy Agency (EEA).
 42. Scora, G. and M. Barth, *Comprehensive modal emission model (CMEM), version 3.01: User's guide*. University of California, 2006.
 43. Trachanatzi, D., et al., *A firefly algorithm for the environmental prize-collecting vehicle routing problem*. Swarm and Evolutionary Computation, 2020. **57**: p. 100712.
 44. Arthur, D. and S. Vassilvitskii, *k-means++: The advantages of careful seeding*. Society for Industrial and Applied Mathematics, 2007: p. 1027-1035.
 45. Kodinariya, T. and P. Makwana, *Review on Determining of Cluster in K-means Clustering*. International Journal of Advance Research in Computer Science and Management Studies, 2013. **1**: p. 90-95.
 46. Goetschalckx, M. and C. Jacobs-Blecha, *The vehicle routing problem with backhauls*. European Journal of Operational Research, 1989. **42**(1): p. 39-51.
 47. Lourenço, H.R., O.C. Martin, and T. Stützle, *Iterated Local Search: Framework and Applications*, in *Handbook of Metaheuristics*, M. Gendreau and J.-Y. Potvin, Editors. 2010, Springer US: Boston, MA. p. 363-397.
 48. Lin, S., *Computer Solutions of the Traveling Salesman Problem*. Bell System Technical Journal, 1965. **44**(10): p. 2245-2269.
 49. Ruiz, E., et al., *Solving the open vehicle routing problem with capacity and distance constraints with a biased random key genetic algorithm*. Computers & Industrial Engineering, 2019. **133**: p. 207-219.
 50. Zografos, K.G. and K.N. Androutsopoulos, *A heuristic algorithm for solving hazardous materials distribution problems*. European Journal of Operational Research, 2004. **152**(2): p. 507-519.
 51. Zografos, K.G. and K.N. Androutsopoulos, *A decision support system for integrated hazardous materials routing and emergency response decisions*. Transportation Research Part C: Emerging Technologies, 2008. **16**(6): p. 684-703.
 52. Androutsopoulos, K.N. and K.G. Zografos, *Solving the bicriterion routing and scheduling problem for hazardous materials distribution*. Transportation Research Part C: Emerging Technologies, 2010. **18**(5): p. 713-726.
 53. Androutsopoulos, K.N. and K.G. Zografos, *A bi-objective time-dependent vehicle routing and scheduling problem for hazardous materials distribution*. EURO Journal on Transportation and Logistics, 2012. **1**(1-2): p. 157-183.
 54. Bula, G.A., et al., *Variable neighborhood search to solve the vehicle routing problem for hazardous materials transportation*. Journal of Hazardous Materials, 2017. **324**: p. 472-480.

55. Yazdani, M., et al., *Reliability estimation using an integrated support vector regression – variable neighborhood search model*. Journal of Industrial Information Integration, 2019. **15**: p. 103-110.
56. Bula, G.A., et al., *Bi-objective vehicle routing problem for hazardous materials transportation*. Journal of Cleaner Production, 2019. **206**: p. 976-986.
57. Bérubé, J.-F., M. Gendreau, and J.-Y. Potvin, *An exact ϵ -constraint method for bi-objective combinatorial optimization problems: Application to the Traveling Salesman Problem with Profits*. European Journal of Operational Research, 2009. **194**(1): p. 39-50.
58. Zandieh, M. and B. Aslani, *A hybrid MCDM approach for order distribution in a multiple-supplier supply chain: A case study*. Journal of Industrial Information Integration, 2019. **16**: p. 100104.
59. Yan, C., et al., *A new method for real-time evaluation of urban traffic congestion: a case study in Xi'an, China*. Geocarto International, 2020. **35**(10): p. 1033-1048.
60. Zeng, G., D. Duan, and P. Zhang, *Comprehensive evaluation of Jing-Jin-Ji region coordinated development based on principal components analytical method*. Sci Technol Prog Policy, 2008. **25**(9): p. 44-49.
61. Rui, Z., et al., *A quantitative oil and gas reservoir evaluation system for development*. Journal of Natural Gas Science and Engineering, 2017. **42**: p. 31-39.
62. Rahman, S.M.A., et al., *Impact of idling on fuel consumption and exhaust emissions and available idle-reduction technologies for diesel vehicles – A review*. Energy Conversion and Management, 2013. **74**: p. 171-182.
63. Shen, G. and D. Ye, *A distance-based spectral clustering approach with applications to network community detection*. Journal of Industrial Information Integration, 2017. **6**: p. 22-32.