Omega 130 (2025) 103164

Contents lists available at ScienceDirect

Omega

journal homepage: www.elsevier.com/locate/omega

Two-echelon collaborative many-to-many pickup and delivery problem for agricultural wholesale markets with workload balance

Jian Li [a], Lu Cang [a], Yisheng Wu [b], Zhaotong Zhang [a,*]

[a] College of Information Management, Nanjing Agricultural University, Nanjing 210031, China
[b] School of Economics & Management, Nanjing Institute of Technology, Nanjing, China

ARTICLE INFO

Keywords:
Two-echelon vehicle routing problem
Many-to-many pickup and delivery problem
Workload balance
Collaboration
Two-stage iterative algorithm

ABSTRACT

In the context of cooperation distribution among multiple wholesale markets, each customer can place orders with multiple agricultural wholesale markets, and each agricultural wholesale market can supply different products. To improve their distribution efficiency and avoid traffic congestion, cargoes need to be transshipped among collaborative distribution of different agricultural wholesale markets owing to their product heterogeneity. Each agricultural wholesale market undertakes the cost of completing the task for the reassigned customers of finding a balance among them, respectively. Fairness of allocation is achieved through workload balance and individual rationality constraints instead of monetary transfer payments. Aimed at the collaboration problem, a two-echelon collaborative many-to-many pickup and delivery problem with workload balance (2E-MPDP-WB) is presented in this study. A mixed-integer programming model for 2E-MPDP-WB is established, and a two-stage iterative algorithm combining branch-and-bound algorithm and adaptive large neighborhood search algorithm is proposed according to the problem structure. Some valid inequalities are also proposed. Finally, computational experiments show the correctness of the model and effectiveness of the algorithm, and sensitivity analysis is performed from four aspects, namely, costs before and after collaboration, workload balance, market geographical distribution, and demand mixing degree. The findings provide management insights for the collaborative distribution of multiple wholesale markets.

1. Introduction

Chinese agriculture is characterized by small-scale and decentralized production, whereas agricultural product sale is characterized by large-scale, cross-regional markets involving long-distance transportation and extensive distribution. To effectively reduce the cost of agricultural distribution, agricultural wholesale markets have the role of handling the quick collection and distribution of fresh agricultural products. According to data released by the China Agricultural Wholesale Market Association, the national agricultural wholesale market reaching a turnover of 5.81 trillion yuan and a volume of 980 million tons in 2021, accounting for 36.7 % of the agricultural product turnover of China that year. Therefore, the agricultural product wholesale market is still one of the main channels for the country's agricultural product circulation. As people's living standard continue to improve, they are putting forward higher requirements for the distribution efficiency and quality of agricultural products, and the government is actively implementing relevant policies to improve the construction and operation mode of the agricultural product wholesale market. In May 2019, the "Notice on Promoting the Interconnection of Agriculture and Commerce to Improve the Supply Chain of Agricultural Products" by the Ministry of Commerce of China mentioned that the government supports agricultural product circulation enterprises in building or renovating agricultural product retail markets and developing new modes with sharing resource, such as joint procurement, shared warehouse, and distribution.

Owing to the differences of products in various wholesale markets, customers may place orders from multiple wholesale markets simultaneously. Distributing agricultural products independently by each market will easily lead to overlapping transportation routes, increased urban congestion, and so forth. Therefore, the collaborative distribution of agricultural wholesale markets is developed to reduce the logistics cost of agricultural products.

Given the different products in each wholesale market, the products are transshipped between wholesale markets and then distributed by the

Area - Production Management, Scheduling and Logistics. This manuscript was processed by Associate Editor Singh.

* Corresponding author.
E-mail address: zzt5176@163.com (Z. Zhang).

https://doi.org/10.1016/j.omega.2024.103164
Received 2 January 2024; Accepted 20 July 2024
Available online 6 August 2024

wholesale markets for customers assigned to it to achieve a one-time cargo receipt for the same customer ordering from different wholesale markets. For one agricultural wholesale market, any other agricultural wholesale market may be the cargo source and destination in collaboration transportation. Therefore, transportation demand among the wholesale markets is many-to-many and less-than-truckload. The wholesale markets simultaneous play the roles of depots (warehouses) and satellites (transshipment centers), while depots and satellite are different facilities in classic two-echelon vehicle routing problem (2E-VRP). In the 2E-VRP, it is generally assumed that there is one central depot from which vehicles depart and return. However, in this collaborative transportation, there is no central depot, and each agricultural market has its own vehicles, respectively. Therefore, the network structure studied in this study is different from other 2E-VRPs. Similar to other 2E-VRPs, customer time windows and maximum vehicle travel length are considered to guarantee the quality of food materials. Due to the time windows, the interaction between the upper and lower layers makes the coordination of time, goods, and vehicles difficult. Moreover, considering the different types of goods transported in the distribution and the size of cargo volume in different echelons of the network, heterogeneous vehicles are used in the first and second echelons. In a first-echelon distribution network, the wholesale market is allowed to be visited by multiple vehicles.

Establishing a collaboration mechanism with different partners is also crucial, except for minimizing the cost of collaborative transportation. In multi-partner collaboration, a fair benefit allocation strategy is the key to the success of collaboration. The main way to solve the problem of benefit allocation in collaboration is to allocate benefits through monetary transfer payment after minimizing the total cost of the coalition [1]. However, we adopt non-monetary transfer payment way to realize benefit balance, e.g., workload balance, which is discussed in Matl et al. [2] and Mancini et al. [3]. It is more beneficial under self-operated logistics because it can reduces the loss from idle company-owned vehicles and provides employee satisfaction. Additionally, the traditional approaches by implementing a posteriori gain sharing mechanisms comes also with some challenge. First, a fair allocation solution cannot be obtained for the vehicle routing problems (VRP) since most of them do not have a core solution, and a satisfactory allocation solution can only be obtained by relaxing some of the conditions. Second, participants may hesitate to cooperate as long as they are unaware of the mechanisms adopted and whether they are receiving a fair share from the collaborative operation [4]. Therefore, we adopt a pre-set allocation rule and embed it into route optimization to simultaneously achieve route optimization and a benefit balance. The method can fully compensate the possible losses of each participant in a flexible manner [5,6].

The main contributions of this study are as follows. First, 2E-MPDP-WB is proposed, and the corresponding mathematical model is established by considering network structure, time windows, two-echelon synchronization, heterogeneous vehicles, workload balance, and individual rationality. Some valid inequalities are proposed to strengthening the formulation. Second, the interactions between the first echelon and second echelon of the problem increase the difficulty of solving. To address this, the model is decomposed into two separate sub-models: pickup and delivery problem with many-to-many demands and time deadline (PDPMDTD) and multi-depot vehicle routing problem with time windows (MDVRPTW), which are solved by branch-and-bound algorithm and adaptive large neighborhood search (ALNS) algorithm, respectively. Then, a two-stage iterative algorithm (TI) is proposed to integrate the two methods. Third, minimizing cost and benefit balance are simultaneously integrated to solve the benefits of both the coalition and the individual partners instead of optimization first and then allocation second. Fairness of allocation is achieved through workload balance and individual rationality constraints. Finally, computational results show the effectiveness of the 2E-MPDP-WB formulation, proposed inequalities, and the TI algorithm, and some valuable management insights are provided for collaborative distribution in multiple wholesale markets. For example, with the gradual relax of workload balance constraints, the total cost of coalition shows a trend of the decline. However, the decline trend will terminate when the number of allowable deviation customers reaches a small increment threshold.

The rest of this paper is organized as follows. Section 2 discusses the literature review related to 2E-MPDP-WB. Section 3 defines and formulates the problem and proposes the valid inequalities and splitting models. Section 4 presents the TI algorithm in detail. Section 5 discusses the numerical experiments and identifies management insights. Lastly, Section 6 summarizes our work and gives future research directions.

2. Literature review

In a two-echelon logistics network, collaboration and sharing of logistics resources among multiple participants can not only maximize the utilization of resources but also reduce the operating cost of the whole logistics network [7,8]. The 2E-MPDP-WB considers the coordination between two-echelon logistics facilities and collaboration among multiple participants. Therefore, this section focuses the literature review on the two-echelon vehicle routing problem (2E-VRP) and collaborative multi-depot vehicle routing problem (CMDVRP), which considers the benefit allocation among each depot (or enterprise) on the basis of multi-depot vehicle routing problem (MDVRP). In addition, for the broader review on the MDVRP, the reader is refered to Rabbouch et al. [9]. Regarding the two-echelon location-routing problem and multi-depot vehicle routing with pickup and delivery, please see Wang et al. [10,11] and Koç et al. [12], respectively.

2.1. 2E-VRP

The classic 2E-VRP involves first-echelon vehicles transporting cargo from the depot to the satellite, where the cargo is unloaded from the first-echelon vehicle and transferred to the second-echelon vehicle, which subsequently delivers the cargo from the satellite to the customers [13,14]. With the many restrictions affecting vehicle transportation in real life, such as traffic restrictions on the roads, time windows of customers, and vehicle capacity restrictions, an increasing number of studies have taken realistic factors into account in 2E-VRP, resulting in many variants. Sluijk et al. [15] gave a detailed overview of the 2E-VRP. We comment on the key elements related to 2E-VRP as follows.

Wang et al. [16] studied the two-echelon multi-depot collaborative distribution problem, where the first-level distribution is direct transportation and the second-level distribution involves VRP. The freights at each depot are assumed to either be identical or interchangeable. The objective is to minimize the total cost of collaboration distribution and then allocate the cost-savings using the Shapley value method. Dellaert et al. [17] introduced a 2E-VRP in which the heterogeneity of cargo between multiple depots is considered. In that case, each demand starts from a specific depot, transfers are done using intermediate facilities and are supposed to be delivered to a specific customer. Jia et al. [18] introduced the multi-commodity two-echelon vehicle routing problem with satellite synchronization (MC2E-VRPSS) to determine the cost minimization route of the two echelons. MC2E-VRPSS involves the transportation from two depots to satellites on the first echelon and the deliveries from satellites to final customers on the second echelon. Each customer has a deadline and two commodity demands from two distinct depots. Gu et al. [19] stipulated that each depot should distribute a variety of commodities, which must be sent from suppliers to satellites by direct transportation and transshipped by multiple satellites to customers by vehicle routing. Li et al. [20] considered the routing problem on the second level as a one-to-many problem, while the routing problem on the first level is based on bidirectional full truckload. The Clarke and Wright savings heuristic algorithm improved by a local search phase is adopted for the problem. Dellaert et al. [21] developed a branch-and-price–based algorithm to solve the large 2E-VRP with time window. The first-echelon routing problem is a multi-depot capacitated VRP, and the second echelon routing problem is a multi-depot capacitated VRP with time windows. Anderluh et al. [22] proposed a multi-objective two-echelon VRP with vehicle synchronization and "gray zone" customers. Inner-city center deliveries are performed by small vehicles because of access restrictions, while deliveries outside this area are carried out by conventional vehicles for economic reasons. The goods are transferred from the first echelon to the second echelon through vehicle synchronization at all echelons.

At present, there are relatively few articles in 2E-VRP that consider the pickup and delivery operations of customers. It is assumed that every customer and satellite has a demand for pickup and delivery, and each echelon corresponds to the VRP with pickup and delivery to minimize the total transportation cost, travel time, or fuel consumption [23–26]. Wang et al. [27] investigated a two-echelon logistics pickup and delivery network optimization problem that considers integrated cooperation and transportation fleet sharing and assumes delivery before pickup and where satellites consist of delivery centers and pickup centers. The computational results show that delivery centers and pickup centers can achieve more cost-savings when they collaborate than when they operate independently Li et al. [28] expressed the sending–receiving relationship between the sender of the origin satellite and the receiver of the destination satellite as a many-to-many demand delivery problem when studying the long-distance transportation between first-echelon satellites. On the second echelon, reception from the sender to the origin satellite and transmission from the destination satellite to the receiver are considered. They divided express transportation into three stages: pickup stage, intercity linehaul, and delivery stage. The three stages are handled as the VRP with time window. Different from Li et al. [28], our work completes pickup and delivery at a satellite or depot and

Table 1
Characteristics of 2E-VRP in relevant literature.

Literature	The first echelon of two echelons				Heterogeneous vehicle	The No. of vehicles	Heterogeneity of goods among multiple depots	Customer time window	Workload balance	Individual rationality	Benefit allocation method	
	Single depot	Multiple depots	Non-PDP	PDP							Monetary transfer payment	Non-monetary transfer payments
Anderluh et al. [22]	√		VRP		√							
Belgin et al. [23]	√		VRPDP		√							
Dellaert et al. [21]		√	MDVRP		√			√				
Dellaert et al. [17]		√	MDVRP		√		√	√				
Gu et al. (2022)		√	DD		√							
Jia et al. [18]		√	MDVRPTW		√	√	√	√				
Li et al. [20]	√		VRPSP		√	√		√				
Li et al. [28]	√		VRPTW		√			√				
Li, Xu & Sun. [24]	√		VRPDP		√	√						
Liu and Liao [7]	√		VRP		√						√	
Paul et al. [25]	√		VRPDP		√	√		√				
Wang et al. [8]	√		DD		√						√	
Wang et al. [16]		√	DD		√						√	
Wang et al. [27]	√		VRP		√						√	
Wang et al. [30]	√		DD		√						√	
Wang et al. [29]	√		VRP		√	√		√		√	√	
Zhou et al. [26]	√		VRPDP		√			√				
Our work		√		√	√	√	√	√	√	√		√

Note: DD indicates direct delivery. VRP indicates vehicle routing problem. VRPSP indicates VRP with split pick-ups. VRPDP indicates VRP with delivery and pickup. VRPTW indicates VRP with time window. MDVRP indicates multi-depot VRP. MDVRPTW indicates MDVRP with time window.

handle it as PDP with many-to-many demand and time deadline. At the same time, the focus of our work is to consider the PDP under the background of multi-partner collaboration, whereas Li et al. [28] focuses on the role of satellite bi-synchronization from the perspective of single enterprise.

Establishing a collaboration mechanism in 2E-VRP is important to reduce logistics costs effectively and improve distribution efficiency for enterprise participants. Wang et al. [29] studied the two-echelon collaborative multi-depot multi-period vehicle routing problem by sharing logistics resources in multiple service time cycles. They determined a solution that can minimize logistics operation cost, service waiting time, and number of vehicles and then designed a cost allocation mechanism. Wang et al. [30] proposed a three-stage approach to solve a two-echelon collaborative multi-center VRP that can minimize both operational costs and CO_2 emissions and derived a law that is favorable to the alliance. Liu and Liao [7] studied a two-echelon collaborative waste collection vehicle routing problem (2E-CWCVRP) considers cooperation and profit distribution among collection network participants to reduce waste collection costs and achieve sustainable urban development.

In the two-echelon VRP, most works focus on minimizing the overall cost of independent operation, but only a few consider collaborative operation. There is still a lack of consideration for the multiple needs of customers in the many-to-many relationship between customers and depots as well as between depots. Although some literature has addressed the vehicle routing problem with delivery and pickup (VRPDP) in 2E-VRP, consideration for the pickup and delivery problem (PDP) for 2E-VRP is still lacking. In terms of cooperative benefit allocation, the focus has mainly been on achieving fairness through monetary transfer payments, but research on achieving allocation through workload balance and individual rational constraints remains scarce. The characteristics of the relevant literature are shown in Table 1.

2.2. CMDVRP

With the rapid development of the logistics industry, scholars have gradually deepened their research on multi-depot transportation networks [31–33]. Compared with traditional VRP, multi-depot vehicle routing problem (MDVRP) enables vehicles to start from multiple depots, provide services to a series of customers according to a certain route plan, and then return to their original depots. In practice, the time window has a significant impact on customer satisfaction regarding the quality of logistics services. As highlighted in the review by Rabbouch, Mrahi, and Saadaoui [9], MDVRPTW has garnered extensive attention. With the development of the sharing economy, CMDVRP has attracted the attention of both theoretical researchers and practitioners. As a variant of MDVRP, CMDVRP is usually viewed as a combination of MDVRP and a benefit allocation problem among depots (enterprises), and its optimization is based on a single-echelon network rather than a two-echelon network.

Wang et al. [34] focuses on establishing a collaborative network optimization model to coordinate the pickup and delivery services between pickup and delivery depots and determine the optimal routes with reduced operating cost through logistics resource sharing. Soriano, Gansterer, and Hartl [35] studied the multi-depot vehicle routing problem with profit fairness under the background of cooperation, which adds a fairness objective function to the classical cost minimization function, and discussed the effect of integration fairness in the optimization process. Xue [36] investigated the impact of carbon emissions on multi-depot collaborative pickup and delivery network with time window and transshipment and proposed a two-stage heuristic method to solve the problem. The results show that sharing transshipment facilities among different warehouses can significantly reduce the emission cost and promote sustainable development. In most of the above studies, the products in different depots are assumed to be the same or substitutable and only transfer orders are required. However, Zhang et al. [37] considered the different product types of different depots in a heterogeneous multi-depot collaborative vehicle routing problem (HMCVRP) and implemented product transfer by using depots and customers as access nodes on the vehicle routes. A branch-and-bound algorithm based on benders was also developed. At the same time, the authors revealed the importance of depot location and put forward a well-designed cost allocation method.

All the above studies about CMDVRP are designed based on a single-echelon network structure, whereas our work studies a two-echelon network structure. Moreover, most of the literature assumes that the products at the depots are either identical or substitutable except for Zhang et al. [37] and that monetary transfer payments are used for allocation benefits. The main differences between our work and Zhang et al. [37] are as follows: the transshipment mode considered by Zhang et al. [37] is different from that considered in our work. Zhang et al. [37] studied the single-echelon network, whereas our work examines the two-echelon network by considering the customer time window. In addition, the cost allocation in Zhang et al. [37] involves cost allocation that is carried out separately after optimization, whereas this study integrates benefit allocation into transportation optimization.

3. Problem description and mathematical formula

3.1. Problem description

The application scenario considered in this study is that multiple wholesale markets for agricultural products collaborate horizontally to distribute food materials such as fish, meat, fruits, and vegetables to surrounding customers (e.g., hotels, restaurants, stores). The products of each wholesale market are heterogeneous, such as aquatic products, meat and poultry, and fruits and vegetables. Customers contact each wholesale market through an online trading platform or telephone to place orders, and the wholesale market distributes food materials according to the daily time requirements of customers to ensure the freshness of food materials. The objective is to minimize the cost of the alliance and each partner and maintain fairness among the partners as much as possible.

Fig. 1 shows the two-echelon logistics distribution network before and after collaboration. Fig. 1(a) shows the independent distribution network of multiple wholesale markets for agricultural products before collaboration, where each wholesale market only served its own customers and independently distributed a type of food material. For example, the aquatic product wholesale market only distributed aquatic products to customers. Customers generally have a demand for many kinds of food materials. There are numerous overlapping transportation routes between wholesale markets, resulting in a considerable and unnecessary waste of resources. In addition, there are some customers who are far away from the wholesale market, making long-distance transportation inevitable. After the formation of collaboration among multiple wholesale markets, according to the location and shared orders of different wholesale markets, first, the food materials are exchanged between wholesale markets by trucks to complete the gathering of different food materials on the first echelon, and then the customers in the same area are distributed uniformly by their own vehicles on the second echelon, as shown in Fig. 1(b). Furthermore, multiple wholesale markets jointly decide the transportation route by sharing information, which can effectively reduce the total number of vehicles used.

To describe the two-echelon collaborative logistics network more clearly, the two-echelon logistics network for three wholesale markets are decomposed as an example, as shown in Fig. 2.

(1) According to the classification of the PDP by Berbeglia et al. [38] , the first-echelon distribution problem is equivalent to the PDPMDTD, which aggregates all types of food materials to each wholesale market through the exchange of food materials between wholesale markets. Any node can be the source or

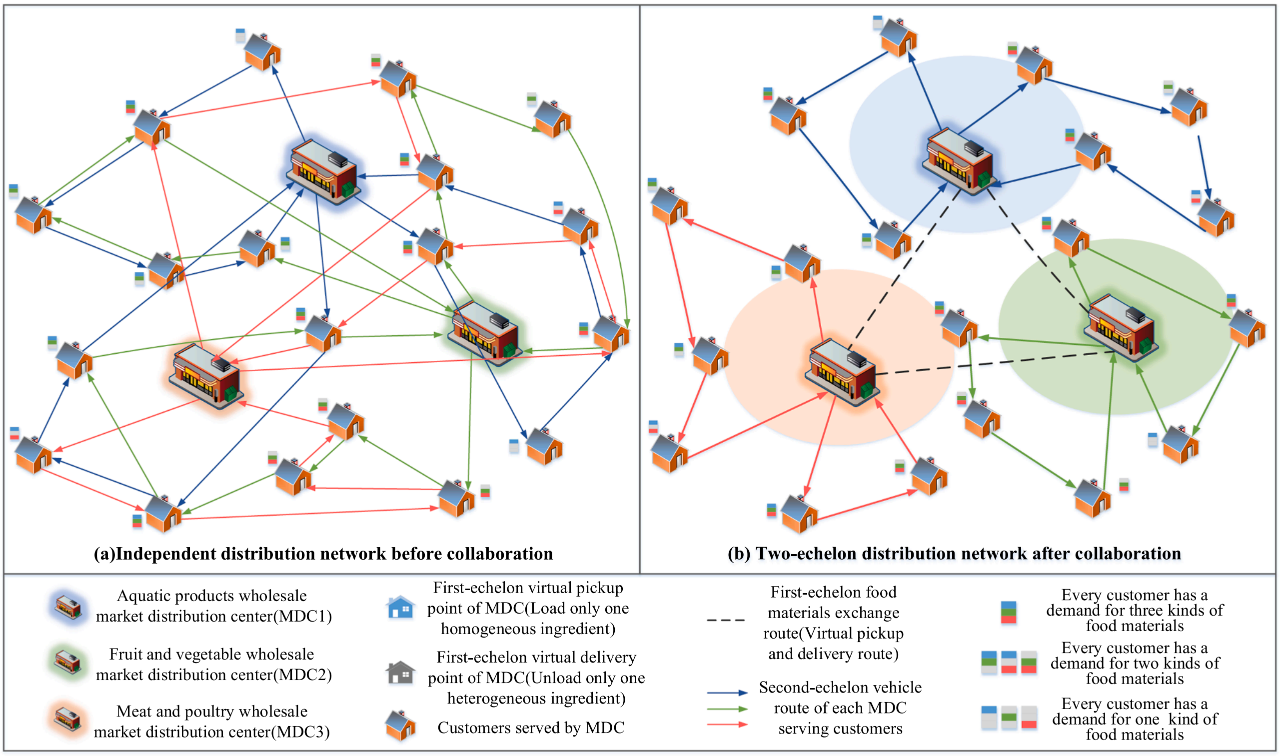

Fig. 1. Two-echelon logistics network before and after collaboration.

destination of any commodity, and the commodity can be picked up from one of many locations and delivered to one of several locations. In a first-echelon distribution network, the same wholesale market is allowed to be visited by multiple vehicles. Therefore, a set of pickup and delivery virtual nodes is set up for each wholesale market to enable vehicles to visit at most one node at a time. Then, the problem is transformed into a one-to-one PDP through virtual nodes. As shown in Fig. 2(b), the first echelon consists of 3 wholesale market distribution centers (MDCs) and 12 virtual pickup and delivery nodes. Each wholesale market has 4 virtual pickup and delivery nodes, whose location coordinates are the same as the parking lot location of the corresponding wholesale market, and the number of virtual pickup and delivery nodes is given according to the number of remaining wholesale markets. Taking the aquatic product wholesale market as an example, the coordinate nodes in the blue area are all the same. Two virtual nodes are used to load aquatic products needed by two other wholesale markets. The remaining two virtual nodes are used to unload the other two kinds of food materials needed by the aquatic product wholesale market. To sum up, in the first-echelon network, each wholesale market picks up food materials from other wholesale markets according to the needs of assigned customers. Each virtual node only loads or unloads food materials once. The vehicle departs from a certain wholesale market and returns to the same wholesale market, finally realizing the exchange of food materials on the first echelon.

(2) The second-echelon distribution problem is equivalent to the MDVRPTW. What is to be achieved here is to deliver the food materials consolidated in each wholesale market to their respective customers. As shown in Fig. 2(c), the second echelon consists of three wholesale MDCs and their corresponding customers. The wholesale MDCs receive food materials according to the assigned customer demand, ensuring timely delivery within designated time windows and striving to minimize costs. Each customer may have a different demand for different food materials, as represented by a three-colored rectangle in Fig. 2(c). If a customer needs a certain food material, the corresponding color will be displayed; otherwise, it will be gray.

(3) The synchronization of two-echelon transportation is considered, that is, the connection of cargo volume and time. For the cargo volume, the demand of customers served by each MDC on the second echelon is calculated first and then reflected on the model on the first echelon. For the two-echelon time, once the second-echelon route of each MDC is determined, the departure time of the second-echelon vehicles is decided, which determines the latest arrival time of the first-echelon vehicles. In this study, the departure time of the earliest departing vehicle in each MDC's second-echelon vehicles is regarded as the latest time when the first-echelon vehicles arrive at the wholesale market and unload all food materials. This is to ensure that the second-echelon vehicles will leave after the first-echelon vehicles complete the unloading service.

(4) Workload balance and individual rationality are required to maintain fairness among the partners. Before collaboration, each MDC has its own set of customers. After collaboration, customers are shared and each MDC is equivalent to having a subset of total customers. The MDC investigated in this study comes from different enterprises. Therefore, to prevent enterprises with large business volume from undertaking a small workload and enterprises with small business volume from undertaking a large workload, this study requires that the number of customers assigned to each MDC should not be less than the number of customers it originally served minus the maximum allowable number. To a certain extent, this requirement can ensure that the customers in each wholesale market do not lose too much and maintain the workload balance. The purpose of collaboration is to minimize the cost, but the minimized total cost does not mean minimized individual cost. Therefore, individual rationality requirements are added to the model. The cost of each MDC after collaboration cannot be higher than that of its independent

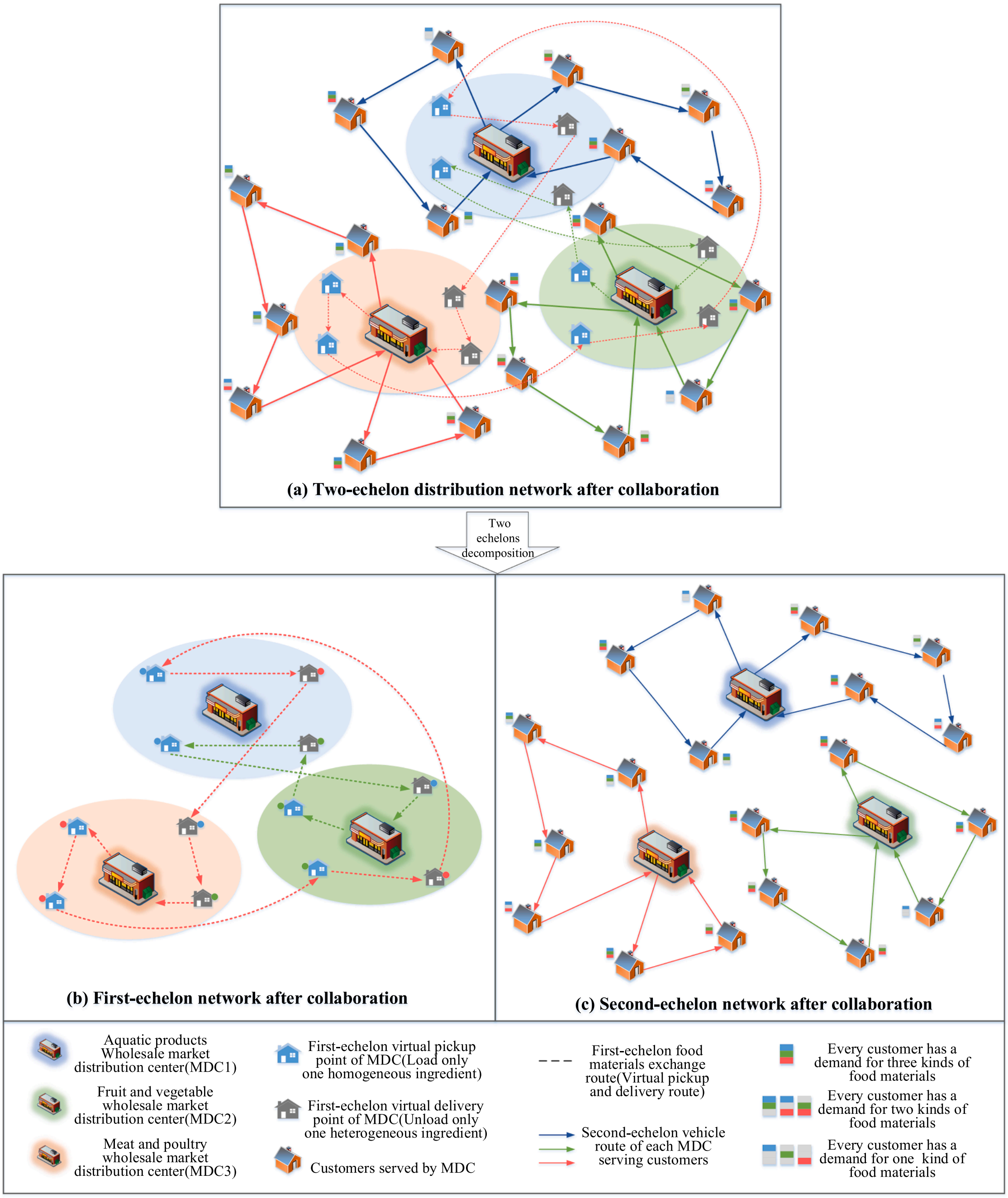

Fig. 2. Exploded view of the logistics distribution of the first and second echelons.

distribution to maintain the stability of collaboration. Each wholesale market undertakes the cost of completing the task for the reassigned customers, respectively, including transportation costs for vehicles and handling fees at this wholesale market.

The implementation of 2E-MPDP-WB depends on the following assumptions:

(1) The vehicle capacities and speeds of the first and second echelons are different, and the vehicle speed of each echelon is a constant (i.e., not varied with time).
(2) Each vehicle has multiple compartments to transport all types of food materials.
(3) The loading and unloading times of vehicles in each wholesale market are the same.
(4) The types and quantities of food materials required by customers are different, and one customer corresponds to one order in one wholesale market.
(5) Each wholesale market corresponds to one type of food material.
(6) The capacity of the wholesale market is large enough, and transshipment is not restricted.
(7) Customers place separate orders in different wholesale markets, and each order cannot be split.

3.2. Mathematical formulation for 2E-MPDP-WB

In this study, a mixed-integer linear programming model is proposed to describe 2E-MPDP-WB, which aims to find the travel route with the minimum total cost. The route optimization and benefit balance are integrated into a model. By solving the 2E-MPDP-WB model, the routes of two-echelon collaborative vehicles can be determined to minimize the costs for the individual and the alliance. The model constraints are divided into four main categories: constraints for the second echelon, constraints for the connection of the two echelons, constraints for the first echelon, and other constraints related to variables.

The definitions of parameters and decision variables are summarized in Table 2.

The 2E-MPDP-WB is formulated as follows (i.e., Model I).

(Model I)

The objective function:

$$min\ Z = b_1\left(\sum_{h\in K_1} uz_h\right) + b_2\left(\sum_{k\in K_2} lz_k\right)$$

$$+a_1\left(\sum_{i\in V_1}\sum_{j\in V_1}\sum_{h\in K_1} d^1_{ij}x_{hij}\right) + a_2\left(\sum_{i\in V_2}\sum_{j\in V_2}\sum_{k\in K_2} d^2_{ij}y_{kij}\right)$$

$$+\sum_{m\in V_D}\sum_{r\in R} DM_{mr}L_m - \sum_{m\in V_D} DM_{mm}L_m \quad (1)$$

The objective function (1) minimizes the total cost, which consists of three parts, namely, fixed cost of vehicles, transportation cost of vehicles, and handling (transfer) cost in wholesale MDCs, where the first item is the fixed cost of vehicles on the first echelon, the second item is the fixed cost of vehicles on the second echelon, the third item is the transportation cost of vehicles on the first echelon, the fourth item is the transportation cost of vehicles on the second echelon, the fifth item is the handling cost of all food materials in all wholesale MDCs, and the sixth item is the handling cost of food materials owned by a certain wholesale MDC itself. i) Constraints for the second echelon.

$$\sum_{j\in V_C} y_{kmj} = \sum_{j\in V_C} la_{kj}, \quad \forall m \in V_D, \forall k \in K_2^m \quad (2)$$

$$y_{kim} = la_{ki}, \quad \forall m \in V_D, \quad \forall k \in K_2^m, \forall i \in V_C \quad (3)$$

$$\sum_{j\in V_2} y_{kji} = la_{ki} + lb_{ki}, \quad \forall m \in V_D, \forall k \in K_2^m, \forall i \in V_C \quad (4)$$

$$\sum_{j\in V_C} y_{kij} = lb_{ki}, \quad \forall m \in V_D, \forall k \in K_2^m, \forall i \in V_C \quad (5)$$

$$\sum_{k\in K_2} (la_{ki} + lb_{ki}) = 1, \quad \forall i \in V_C \quad (6)$$

$$\sum_{j\in V_C} y_{kmj} = \sum_{j\in V_C} y_{kjm}, \quad \forall m \in V_D, \quad \forall k \in K_2^m \quad (7)$$

$$\sum_{m\in V_D} lu_{mi} = 1, \quad \forall i \in V_C \quad (8)$$

$$lz_k = \sum_{j\in V_C} y_{kmj}, \quad \forall m \in V_D, \quad \forall k \in K_2^m \quad (9)$$

$$lz_k \geq la_{ki} + lb_{ki}, \quad \forall m \in V_D, \forall k \in K_2^m, \forall i \in V_C \quad (10)$$

$$lz_k \geq lu_{mi} - (1 - la_{ki} - lb_{ki}), \quad \forall m \in V_D, \forall k \in K_2^m, \forall i \in V_C \quad (11)$$

$$lz_k \leq lu_{mi} + (1 - la_{ki} - lb_{ki}), \quad \forall m \in V_D, \forall k \in K_2^m, \forall i \in V_C \quad (12)$$

$$y_{kmj} = 0, \quad \forall i \in V_D,\ m \in V_D, i \neq m, \forall k \in K_2^i, \forall j \in V_C \quad (13)$$

$$y_{kii} = 0, \quad \forall i \in V_2, \forall k \in K_2 \quad (14)$$

Constraints (2)–(14) ensure that vehicles visit the corresponding nodes and form a trip. Constraints (2)–(3) indicate that vehicle k starts from MDC_m and then returns to MDC_m, that is, the starting node and ending node of the vehicle are the same. Constraints (4)–(5) indicate

Table 2
Parameters and decision variables.

Parameters	
V	Node set, $V = N \cup V_D \cup V_C$
N	Virtual node set of the first echelon, $N = P \cup D$
V_D	Set of *MDCs*
V_C	Set of customers
P	Set of virtual nodes for pickup on the first echelon, $P = \cup_{m\in V_D} NP_m$
D	Set of virtual nodes for delivery on the first echelon, $D = \cup_{m\in V_D} ND_m$
NP_m	Set of virtual nodes of $MDC_m(m \in V_D)$ for pickup on the first echelon, i.e., pick up the goods from MDC_m
ND_m	Set of virtual nodes of $MDC_m(m \in V_D)$ for delivery on the first echelon, i.e., unloading at MDC_m
$N1_m$	Set of all nodes of each $MDC_m(m \in V_D)$ on the first echelon
V_1	Set of all nodes on the first echelon, $V_1 = V_D \cup N$
V_2	Set of all nodes on the second echelon, $V_2 = V_D \cup V_C$
n_d	Number of *MDCs*, $n_d = \|V_D\|$
n	Number of pickup nodes on the first echelon, $n = \|P\|$
n_c	Number of customers, $n_c = \|V_C\|$
K_1	Set of vehicles on the first echelon, $K_1 = \cup_{m\in V_D} K_{11}^m$
K_1^m	Set of vehicles on the first echelon available at $MDCm(m\in V_D)$
K_2	Set of vehicles on the second echelon, $K_2 = \cup_{m\in V_D} K_{22}^m$
K_2^m	Set of vehicles on the second echelon available at $MDCm(m\in V_D)$
n_1^m	Maximum number of vehicles that $MDC_m(m \in V_D)$ can use at the same time on the first-echelon distribution network
n_2^m	Maximum number of vehicles that $MDC_m(m \in V_D)$ can use at the same time on the second-echelon distribution network
R	Set of kinds of food materials (divided by different types of *MDCs*); each food material corresponds to an MDC, $\|R\| = \|V_D\|$
q_{ir}	Food material $r(r \in R)$ required by customer $i(i \in V_C)$
Q_1	Capacity of vehicles on the first echelon
Q_2	Capacity of vehicles on the second echelon
A_m	Number of customers initially served by $MDC_m(m \in V_D)$
g_m	Maximum allowable deviation number of customers of $MDC_m(m \in V_D)$
d^1_{ij}	Distance of arc $(i,j)(i,j \in V_1)$ on the first echelon, supposing $d^1_{ij} = d^1_{ji}$
d^2_{ij}	Distance of arc $(i,j)(i,j \in V_2)$ on the second echelon, supposing $d^2_{ij} = d^2_{ji}$
v_1	Average speed of vehicles on the first echelon
v_2	Average speed of vehicles on the second echelon
t^1_{ij}	Travel time of vehicles on the first echelon on arc$(i,j)(i,j \in V_1)$, $t^1_{ij} = d^1_{ij}/v_1$
t^2_{ij}	Travel time of vehicles on the second echelon on arc$(i,j)(i,j \in V_2)$, $t^2_{ij} = d^2_{ij}/v_2$
UT	Unloading time of vehicles at each node
LT	Loading time of vehicles at each node
$[e_i, l_i]$	Time window of customer $i(i \in V_C)$, e_i is the lower bound and l_i is the upper bound
CB_m	Independent distribution cost of $MDC_m(m \in V_D)$
L_m	Handling cost of a unit of food material at $MDC_m(m \in V_D)$
a_1	Running cost per unit distance of vehicles on the first echelon
a_2	Running cost per unit distance of vehicles on the second echelon
b_1	Fixed cost for each vehicle on the first echelon to complete a trip
b_2	Fixed cost for each vehicle on the second echelon to complete a trip
M	A large enough positive integer
S_1	Maximum mileage of vehicles on the first echelon
S_2	Maximum mileage of vehicles on the second echelon
Decision variable	**Decision variables of the first and second echelons**
First echelon	
UPX_i	Quantity of food materials loaded by the vehicle at node $i(i \in N)$ on the first echelon
USX_i	Quantity of food materials unloaded by the vehicle at node $i(i \in N)$ on the first echelon
LD_{hi}	Load capacity of vehicle $h(h \in K_1)$ when it reaches node $i(i \in N)$
udt_{hi}	Departure time of vehicle $h(h \in K_1)$ from the $MDC_i(i \in N)$
uat_{hi}	Arrival time of vehicle $h(h \in K_1)$ at node $i(i \in N)$
uz_h	Binary decision-variable, which is 1 if and only if vehicle $h(h \in K_1)$ is used
x_{hij}	Binary decision-variable, which is 1 if and only if vehicle $h(h \in K_1)$ passes through the arc$(i,j)(i,j \in V_1)$

(*continued on next page*)

Table 2 (*continued*)

Parameters	
ua_{hi}	Binary decision-variable, which is 1 if and only if vehicle $h(h \in K_1)$ visits the last node $i(i \in N)$ on the route
ub_{hi}	Binary decision-variable, which is 1 if and only if vehicle $h(h \in K_1)$ visits the intermediate node $i(i \in N)$ on the route
DM_{mr}	Demand of food material $r(r \in R)$ by $MDC_m(m \in V_D)$
Second echelon	
QQ_{mir}	Quantity of food material $r(r \in R)$ in the order of customer $i(i \in N)$ passing through $MDC_m(m \in V_D)$
LQX_{kir}	Food material $r(r \in R)$ unloaded by vehicle $k(k \in K_2)$ at customer $i(i \in N)$
ldt_{km}	Departure time of vehicle $k(k \in K_{22}^m)$ from $MDC_m(m \in V_D)$
lat_{ki}	Arrival time of vehicle $k(k \in K_2)$ at customer $i(i \in V_C)$
lz_k	Binary decision-variable, which is 1 if and only if vehicle $k(k \in K_2)$ is used
y_{kij}	Binary decision-variable, which is 1 if and only if vehicle $k(k \in K_2)$ travels through the arc$(i,j)(i,j \in V_2)$
la_{ki}	Binary decision-variable, which is 1 if and only if vehicle $k(k \in K_2)$ visits the last node $i(i \in V_C)$ on the route
lb_{ki}	Binary decision-variable, which is 1 if and only if vehicle $k(k \in K_2)$ visits the intermediate node $i(i \in V_C)$ on the route
lu_{mi}	Binary decision-variable, which is 1 if and only if customer $i(i \in V_C)$ is assigned to$MDC_m(m \in V_D)$

that the vehicle cannot stay at the customer and should leave after serving the customer. Constraint (6) means that each customer can only be visited once. Constraint (7) indicates that the number of times all vehicles depart from MDC_m is equal to the number of times they return to MDC_m. Constraint (8) indicates that each customer can only be assigned to one *MDC*. Constraint (9) indicates that each vehicle can only depart from one *MDC*. Constraint (10) indicates that vehicle k must be used if it visits customer i. Constraints (11)–(12) indicate that if customer i is visited by vehicle k of MDC_m, then MDC_m must serve customer i. Constraint (13) indicates that the vehicles of one *MDC* cannot depart from another *MDC*. Constraint (14) indicates that vehicles are not allowed to go from one node to itself.

$$LQX_{kir} = q_{ir} * (la_{ki} + lb_{ki}), \quad \forall i \in V_C, \forall r \in R, \forall m \in V_D, \forall k \in K_2^m \tag{15}$$

$$QQ_{mir} = q_{ir} * lu_{mi}, \quad \forall m \in V_D, \forall i \in V_C, \forall r \in R \tag{16}$$

$$DM_{mr} = \sum_{i \in V_C} q_{ir} * lu_{mi}, \quad \forall m \in V_D, \quad \forall r \in R \tag{17}$$

$$\sum_{j \in V_C} \sum_{k \in K_{22}^m} y_{kmj} \le n_2^m, \quad \forall m \in V_D \tag{18}$$

$$\sum_{i \in V_C} \sum_{r \in R} q_{ir} * (la_{ki} + lb_{ki}) \le Q_2, \quad \forall m \in V_D, \forall k \in K_2^m \tag{19}$$

Constraints (15)–(19) indicate the cargo volume relationship, vehicle number, and vehicle capacity requirements of the second echelon. Among them, constraint (15) means that the amount of food material r unloaded by the vehicle at customer i is equal to the demand of customer i for food material r. Constraint (16) means that the amount of food material r passing through MDC_m in the order of customer i is equal to the demand of customer i for food material r, and adding these two constraints can shorten the running time of software CPLEX. Constraint (17) indicates that the demand of MDC_m for food material r is equal to the total demand of the customers served by MDC_m for food material r. Constraint (18) indicates the vehicle number constraint and that the number of vehicles departing from MDC_m does not exceed the maximum number of vehicles owned by MDC_m. Constraint (19) indicates the capacity constraint and that the actual load capacity of the vehicle does not exceed the maximum load capacity of the second-echelon vehicle.

$$ldt_k^m + t_{mj}^2 - M \cdot \left(1 - y_{kmj}\right) \le lat_{kj}, \quad \forall m \in V_D, \quad \forall k \in K_2^m, \forall i \in V_C \tag{20}$$

$$ldt_k^m + t_{mj}^2 + M \cdot \left(1 - y_{kmj}\right) \ge lat_{kj}, \quad \forall m \in V_D, \quad \forall k \in K_2^m, \forall i \in V_C \tag{21}$$

$$lat_{ki} + UT + t_{ij}^2 - M \cdot \left(1 - y_{kij}\right) \le lat_{kj}, \quad \forall m \in V_D, \quad \forall k \in K_2^m, \forall i \in V_C, \quad \forall j \in V_C \tag{22}$$

$$lat_{kj} \ge e_j \cdot \left(la_{kj} + lb_{kj}\right), \quad \forall m \in V_D, \quad \forall k \in K_2^m, \forall j \in V_C \tag{23}$$

$$lat_{kj} \ge l_j \cdot \left(la_{kj} + lb_{kj}\right), \quad \forall m \in V_D, \quad \forall k \in K_2^m, \forall j \in V_C \tag{24}$$

$$\sum_{m \in V_D} ldt_{km} \le M \cdot lz_k, \quad \forall k \in K_2 \tag{25}$$

$$\sum_{i \in V_C} lu_{mi} \ge A_m - g_m, \quad \forall m \in V_D \tag{26}$$

$$\sum_{i \in V_2} \sum_{j \in V_2} d_{ij}^2 y_{kij} \le S_2, \quad \forall m \in V_D, \forall k \in K_2^m \tag{27}$$

Constraints (20)–(27) represent time constraints, the number of customers served, and the maximum travel mileage requirements of vehicles on the second echelon. Constraints (20)–(22) indicate the relationship between the departure time of the second-echelon vehicles from MDC and the arrival time of the customers and ensure the continuity of the route. Constraints (23)–(24) ensure that the vehicle arrives within the time window of the customer. Constraint (25) indicates that if the second-echelon vehicle k is not used, then its departure time is 0. Constraint (26) is used to maintain the workload balance, that is, the number of customers allocated to MDC_m cannot be lower than the minimum number of customers that MDC_m can allow. Constraint (27) limits the maximum travel mileage of vehicles on the second echelon. ii) Constraints for the connection of the two echelons.

$$uat_{hj} + UT - M \cdot (1 - ua_{hj} - ub_{hj}) \le ldt_k^m + M \cdot (1 - lz_k),$$

$$\forall m \in V_D, \forall h \in K_1, \forall k \in K_2^m, \forall j \in ND_m \tag{28}$$

$$UPX_i = USX_{i+n}, \quad \forall i \in P \tag{29}$$

$$USX_j = DM_{sm}, \quad \forall m \in V_D, s \in V_D, s \ne m, \forall i \in NP_m, \ j \in ND_s, j = i + n \tag{30}$$

$$UPX_i = 0, \quad \forall i \in V_1 \backslash P \tag{31}$$

$$USX_i = 0, \quad \forall i \in V_1 \backslash D \tag{32}$$

$$b\left(\sum_{h \in K_{11}^m} uz_h\right) + a_1\left(\sum_{i \in V_1} \sum_{j \in V_1} \sum_{h \in K_1^m} d_{ij}^1 x_{hij}\right)$$

$$+c\left(\sum_{k \in K_{22}^m} lz_k\right) + a_2\left(\sum_{i \in V_2} \sum_{j \in V_2} \sum_{k \in K_2^m} d_{ij}^2 y_{kij}\right)$$

$$+\sum_{r \in R} DM_{mr} L_m - DM_{mm} L_m \le CB_m, \quad \forall m \in V_D \tag{33}$$

Constraints (28)–(33) are the key to connecting the first and second echelons. Constraint (28) means that the time for the first-echelon vehicle to unload all the goods at MDC_m is not later than the time for the second-echelon vehicle to depart from MDC_m, and it is guaranteed that the first-echelon vehicle can arrive at MDC_m before the food materials are delivered (respecting the necessary time for cargo transshipment). Constraints (29)–(30) indicate the connection of cargo volumes of the first and second echelons, aiming to make each virtual delivery node of each wholesale market unload only one kind of food material. Constraint (29) indicates that the pickup volume at the virtual pickup node is equal to the unloading volume at its corresponding delivery node. Constraint (30) indicates that the pickup volume of food materials from MDC_m at the unloading node j of MDC_s is equal to the total demand of all customers served by MDC_s for food materials at MDC_m. Constraints (31)–(32) ensure that each node in all virtual nodes of the first echelon is loaded or unloaded only once and that there is only one kind of food material. In addition, the loading quantity of the unloading node and unloading quantity of the loading nodes are both 0. Constraint (33) embodies the individual rationality. It ensures that the cost of MDC_m after collaboration is less than that of its independent distribution, which shows the effectiveness of collaboration. iii) Constraints for the first echelon.

$$\sum_{j\in N} x_{hmj} = \sum_{j\in N} ua_{hj}, \quad \forall m \in V_D, \forall h \in K_1^m \tag{34}$$

$$x_{him} = ua_{hi}, \quad \forall m \in V_D, \forall h \in K_1^m, \forall i \in N \tag{35}$$

$$\sum_{j\in V_1} x_{hji} = ua_{hi} + ub_{hi}, \quad \forall m \in V_D, \forall h \in K_1^m, \forall i \in N \tag{36}$$

$$\sum_{j\in N} x_{hij} = ub_{hi}, \quad \forall m \in V_D, \forall h \in K_1^m, \forall i \in N \tag{37}$$

$$\sum_{h\in K_1} (ua_{hi} + ub_{hi}) = 1, \quad \forall i \in N \tag{38}$$

$$\sum_{j\in N} x_{hmj} = \sum_{j\in N} x_{hjm}, \quad \forall m \in V_D, \quad \forall h \in K_1^m \tag{39}$$

$$uz_h = \sum_{j\in N} x_{hmj}, \quad \forall m \in V_D, \forall h \in K_1^m \tag{40}$$

$$uz_h \geq ua_{hi} + ub_{hi}, \quad \forall m \in V_D, \forall h \in K_1^m, \forall i \in N \tag{41}$$

$$x_{hmj} = 0, \quad \forall i \in V_D,\ m \in V_D, i \neq m, \forall h \in K_1^i, \forall j \in N \tag{42}$$

$$x_{hii} = 0, \quad \forall i \in V_1, \forall h \in K_1 \tag{43}$$

$$x_{hij} = 0, \quad \forall i \in V_D,\ j \in V_D, \forall h \in K_1 \tag{44}$$

Constraints (34)–(44) ensure that vehicles visit the corresponding nodes and form a trip. Among them, constraints (34)–(35) indicate that the vehicle h starts from and then returns to the MDC_m depot, that is, the starting and ending nodes of the vehicle are the same. Constraints (36)–(37) indicate that the vehicle should leave after serving the pickup and delivery nodes to ensure flow balance. Constraint (38) indicates that each node can only be visited once. Constraint (39) indicates that the number of times all vehicles leave the MDC_m is equal to the number of times they return to the MDC_m. Constraint (40) indicates that each vehicle can only depart from one MDC. Constraint (41) indicates that vehicle h must be used if it visits node i. Constraint (42) indicates that vehicles in one depot cannot depart from another depot. Constraint (43) indicates that the first-echelon vehicle cannot go from one node to itself. Constraint (44) indicates that it is impossible to go from MDC to MDC.

$$\sum_{j\in P\setminus NP_m} x_{hmj} = 0, \quad \forall m \in V_D, \forall h \in K_1^m \tag{45}$$

$$\sum_{j\in D} x_{hmj} = 0, \quad \forall m \in V_D, \forall h \in K_1^m \tag{46}$$

$$\sum_{j\in V_1} x_{hij} = \sum_{j\in V_1} x_{hj,n+i}, \quad \forall i \in P, \quad \forall h \in K_1 \tag{47}$$

Constraints (45)–(47) ensure the order of pickup and delivery. Constraints (45)–(46) indicate that the first node visited by vehicle h from MDC_m depot is the pickup node of MDC_m. Constraint (47) ensures that if the pickup node is visited, then the delivery node must be visited and the visit is performed by the same vehicle.

$$LD_{hi} - USX_i + UPX_i - \mathrm{M}\cdot(1 - x_{hij}) \leq LD_{hj}, \quad \forall h \in K_1, \forall i \in V_1, \forall j \in V_1 \tag{48}$$

$$LD_{hi} - USX_i + UPX_i + \mathrm{M}\cdot(1 - x_{hij}) \geq LD_{hj}, \quad \forall h \in K_1, \forall i \in V_1, \forall j \in V_1 \tag{49}$$

$$LD_{hi} \leq Q_1 \cdot \sum_{j\in V_1} x_{hji}, \quad \forall i \in V_1, \quad \forall h \in K_1 \tag{50}$$

$$LD_{hi} = 0, \quad \forall i \in V_D, \quad \forall h \in K_1 \tag{51}$$

$$\sum_{j\in N}\sum_{h\in K_1} x_{hmj} \leq n_1^m, \quad \forall m \in V_D \tag{52}$$

Constraints (48)–(52) indicate the cargo volume relationship, vehicle number, and vehicle capacity requirements of the first echelon. Constraints (48)–(49) represent the cargo relationship between pickup and delivery nodes. Constraint (50) ensures that the vehicle is not overloaded. Constraint (51) indicates that the load capacity at MDC depot is 0. Constraint (52) indicates that the number of first-echelon vehicles departing from MDC_m does not exceed the maximum number of first-echelon vehicles owned by MDC_m.

$$uat_{hi} \leq uat_{h,i+n}, \quad \forall h \in K_1, \forall i \in P \tag{53}$$

$$udt_{hm} + t_{mj}^1 - M\cdot(1 - x_{hmj}) \leq uat_{hj}, \quad \forall m \in V_D, \forall h \in K_1, \forall j \in N \tag{54}$$

$$udt_{hm} + t_{mj}^1 + M\cdot(1 - x_{hmj}) \geq uat_{hj}, \quad \forall m \in V_D, \forall h \in K_1, \forall j \in N \tag{55}$$

$$uat_{hi} + \frac{UT + LT}{2} + t_{ij}^1 - M\cdot(1 - x_{hij}) \leq uat_{hj}, \quad \forall h \in K_1, \forall i \in N, \forall j \in N \tag{56}$$

$$uat_{hi} + \frac{UT + LT}{2} + t_{ij}^1 + M\cdot(1 - x_{hij}) \geq uat_{hj}, \quad \forall h \in K_1, \forall i \in N, \forall j \in N \tag{57}$$

$$\sum_{m\in V_D} udt_{hm} \leq M\cdot uz_h, \quad \forall h \in K_1 \tag{58}$$

$$\sum_{i\in V_1}\sum_{j\in V_1} d_{ij}^1 x_{hij} \leq S_1, \quad \forall h \in K_1 \tag{59}$$

Constraints (53)–(59) represent the time constraints and maximum travel mileage requirements of vehicles on the first echelon. Constraint (53) ensures that each pickup node is prior to its delivery node. Constraints (54)–(57) are similar to constraints (20)–(22), indicating the relationship between the departure time of the first-echelon vehicles from the depot of MDC and the arrival time of the pickup and delivery virtual nodes. Constraint (58) indicates that if the first-echelon vehicle h is not used, then its departure time is 0. Constraint (59) limits the maximum mileage of the first-echelon vehicles. iv) Other constraints.

$$UPX_i,\ USX_i \geq 0, \quad \forall i \in V_1 \tag{60}$$

$$LD_{hi}, udt_{hm}, uat_{hi} \geq 0, \quad \forall h \in K_1, \forall i \in N, \quad \forall m \in V_D \tag{61}$$

$$QQ_{mir}, LQX_{kir}, DM_{mr}, ldt_{km}, lat_{ki} \geq 0, \quad \forall k \in K_2, \forall m \in V_D, \quad \forall i \in V_C, \forall r \in R \tag{62}$$

$$uz_h, ua_{hi}, ub_{hi} = \{0, 1\}, \quad \forall h \in K_1, \forall i \in N \tag{63}$$

$$x_{hij} = \{0, 1\}, \quad \forall h \in K_1, \quad \forall i \in V_1, \quad \forall j \in V_1 \tag{64}$$

$$lu_{mi}, lz_k, la_{ki}, lb_{ki} = \{0, 1\}, \quad \forall k \in K_2, \forall m \in V_D, \quad \forall i \in V_C \tag{65}$$

$$y_{kij} = \{0, 1\}, \quad \forall k \in K_2, \forall i \in V_2, \forall j \in V_2 \tag{66}$$

Constraints (60)–(62) indicate that the decision variables are nonnegative. Constraints (63)–(66) represent the binary restrictions on the decision variables.

Model I involves the independent distribution cost of each wholesale market (*MDC*), so it is necessary to establish an independent distribution model to solve it. The independent distribution problem of each wholesale market is equivalent to the vehicle routing problem with time window (VRPTW), and the formulation is similar to the second-echelon constraints in Model I. The details are as follows (i.e., Model II).

(Model II)

The objective function:

$$min\ Z' = c\left(\sum_{k\in K} lz_k\right) + a\left(\sum_{i\in V}\sum_{j\in V}\sum_{k\in K} d_{ij}y_{kij}\right) \tag{67}$$

Subject to $(2)-(7),(9)-(10),(14)-(15),(17)-(24),(27).$

Different from the second-echelon constraint in Model I, all the parameters and decision variables in Model II are set for a single wholesale market, which is equivalent to a VRPTW. Among them, the objective function is to minimize the total cost of the wholesale market, including the fixed cost and transportation cost of vehicles. K denotes the set of vehicles owned by this wholesale market. V represents a distribution center {0} and all customers V_C corresponding to the wholesale market. LQX_{ki} represents the amount of food materials unloaded by vehicle k at customer i. q_i represents the demand of customer i for food materials in the market. DM represents the total amount of food materials sent from the market. nk represents the maximum number of vehicles owned by the market. $m, K_{22}^m, V_2, LQX_{kir}, q_{ir}, n_2^m$ in the other constraints are changed to $0, K, V, LQX_{ki}, q_i, nk$, respectively, and lu_{mi} and $m \in V_D$ are deleted at the same time. Since the customers are known when each wholesale market distributes independently, the total amount of food materials delivered is also determined. Constraint (17) is changed to constraint (68), which means that the total amount of food materials delivered from this wholesale market is equal to the unloading amount of vehicles at all customers.

$$DM = \sum_{k\in K}\sum_{i\in V_C} LQX_{ki} \tag{68}$$

3.3. Valid inequalities

As far as describing the problem is concerned, the model in Section 3.2 is complete enough, and the optimal solution can be obtained through the software CPLEX solver. However, because there are many variables in this model and the feasible region is large, it takes a long time even for some small-size instances to get the optimal solution. Therefore, by analyzing the nature of the problem and aiming at the different characteristics of the two-echelon routes in this model, this section constructs the following valid inequalities. The purpose is to tighten the feasible region of the model and speed up the solution without affecting any feasible solution of the model in this study.

Valid inequalities for the second-echelon route:

$$\sum_{k\in K_2}\sum_{m\in V_D}\sum_{j\in V_C} y_{kmj} \geq \sum_{k\in K_2}\sum_{i\in V_2}\sum_{j\in V_2} d_{ij}^2 y_{kij} \Big/ S_2 \tag{69}$$

$$\sum_{k\in K_2}\sum_{m\in V_D}\sum_{j\in V_C} y_{kmj} \geq \sum_{i\in V_C}\sum_{r\in R} q_{ir} \Big/ Q_2 \tag{70}$$

$$\sum_{j\in V_C}\sum_{k\in K_2} y_{kmj} \geq \sum_{r\in R} DM_{mr} \Big/ Q_2, \forall m \in V_D \tag{71}$$

Constraint (69) limits the lower of the number of vehicles used on the second echelon. That is, the number of vehicles actually used is not less than the number of vehicles when each vehicle runs the maximum mileage. From the point of view of all vehicles on the second level, constraint (70) indicates that the total number of vehicles used on the second echelon is not less than the number of vehicles when all demands are fully loaded. From the perspective of the number of vehicles in each wholesale market, constraint (71) indicates that the number of vehicles used in the wholesale market is not less than the number of vehicles when all the demands in the wholesale market are fully loaded.

Valid inequalities for the first-echelon route:

$$x_{hij} = 0, \forall m \in V_D, h \in K_1, i \in NP_m, j \in ND_m \tag{72}$$

$$x_{hjm} = 0, \forall m \in V_D, h \in K_1^m, j \in P \tag{73}$$

$$x_{h,i+n,i} = 0, \quad \forall i \in P, h \in K_1 \tag{74}$$

$$\sum_{i\in ND_m}\sum_{j\in NP_m} x_{hij} \leq 1, \forall m \in V_D, h \in K_1 \tag{75}$$

$$\sum_{i\in N1_m}\sum_{j\in N1_n} x_{hij} \leq 1, \forall m \in V_D, n \in V_D, m \neq n, h \in K_1 \tag{76}$$

Constraint (72) means that the vehicle does not go from the pickup node to the unloading node for the same *MDC*, which is based on the fact that the goods picked up by the vehicle from its own home do not need to be sent to itself. Constraint (73) indicates that it is impossible for vehicles in each wholesale market to return to the depot from the pickup node. Specifically, the vehicle needs to return to the depot from the unloading node. Constraint (74) means that the vehicle should not go from the corresponding unloading node to the pickup node first. That is, it is guaranteed that the goods must be picked up first and then unloaded in the pickup and delivery route. Constraints (75) and (76) are considered from the view of an arc, aiming at reducing useless routes to tighten the feasible region. Constraint (75) indicates that for the same *MDC*, the arc of each vehicle from the unloading node to the pickup node does not exceed 1. Constraint (76) indicates that the connecting arc between different *MDCs* does not exceed 1. For the validation of these valid inequalities, see Section 5.2.2 for details.

3.4. Model decomposition

The mathematical formulation for 2E-MPDP-WB becomes more complicated because this model explicitly considers two kinds of problems: the PDPMDTD and the MDVRPTW. Owing to the NP-hard characteristics of 2E-MPDP-WB, it is also difficult to obtain the optimal solutions of all problems by using mathematical models, especially for medium- and large-size instances. Therefore, we consider splitting the 2E-MPDP-WB model into two sub-models: Model III and Model IV. Then, the respective algorithms are designed for each of the two sub-models, and the iterative algorithm is employed to integrate them so as to quickly find a better solution to the medium- and large-size instances. The TI algorithm is described in Section 4.

(Model III)

The objective function:

$$min\ Z2 = c\left(\sum_{k\in K_2} lz_k\right) + a_2\left(\sum_{i\in V_2}\sum_{j\in V_2}\sum_{k\in K_2} d_{ij}^2 y_{kij}\right) \tag{77}$$

Subject to $(2)-(27),(62),(65)-(66).$

Optimization of Model III corresponding to the route of the second echelon (i.e., the MDVRP with time windows). The objective function is to minimize the total cost of the second echelon, including the fixed cost and traveling cost of vehicles. We adopt the ALNS algorithm described in Section 4.2 to solve Model III.

(Model IV)

The objective function:

$$\begin{aligned} min\ Z1 &= b\left(\sum_{h\in K_1} uz_h\right) + a_1\left(\sum_{i\in V_1}\sum_{j\in V_1}\sum_{h\in K_1} d^1_{ij}x_{hij}\right) \\ &+ \sum_{m\in V_D}\sum_{r\in R} DM_{mr}L_m - \sum_{m\in V_D} DM_{mm}L_m \end{aligned} \tag{78}$$

Subject to $(28) - (61), (63) - (64)$.

Among them, constraints (28) and (33) are modified as constraints (28a) and (33a), respectively, where l_m indicates the latest departure time of the first-echelon vehicles, and f_2^m represents the cost of the wholesale market MDC_m on the second echelon.

$$uat_{hj} + UT \le l_m + M\cdot(1 - ua_{hj} - ub_{hj}), \quad \forall m \in V_D, \forall h \in K_1, \forall i \in ND_m \tag{28a}$$

$$\begin{aligned} & b\left(\sum_{h\in K_{11}^m} uz_h\right) + a_1\left(\sum_{i\in V_1}\sum_{j\in V_1}\sum_{h\in K_{11}^m} d^1_{ij}x_{hij}\right) + f_2^m \\ & + \sum_{r\in R} DM_{mr}L_m - DM_{mm}L_m \le CB_m, \quad \forall m \in V_D \end{aligned} \tag{33a}$$

Model IV corresponds to the optimization of the first-echelon route. The objective function (78) is to minimize the total cost of the first echelon, including the fixed cost of vehicles, transportation cost of vehicles, and handling (transshipment) cost of wholesale MDCs. Constraint (28a) indicates that each wholesale market MDC_m on the first echelon meets the requirement of the cargo transshipment time between the two echelons. Constraint (33a) guarantees the cost assigned to each wholesale market is less than its independent distribution cost. The values of l_m and f_2^m are obtained from the calculation results of the second echelon, by which the first-echelon model is connected with the second-echelon model. The route optimization problem of the first echelon is equivalent to the PDPMDTD. As the number of nodes involved in the first echelon is small, we employ the branch-and-bound algorithm by utilizing software CPLEX introduced in Section 4.3 to get the optimal solution quickly.

4. Two-stage iterative algorithm

The transportation service in the first-echelon distribution and the customer service in the second-echelon distribution interact with each other in 2E-MPDP-WB. Therefore, we propose a TI algorithm to solve it.

The whole process of the TI algorithm is given in Section 4.1, and then the ALNS for the second-echelon sub-model is introduced in Section 4.2. Next, the algorithm for the first-echelon sub-model is discussed in Section 4.3. Finally, the iterative integration process of the two-echelon algorithm is discussed in Section 4.4.

4.1. Framework of the two-stage iterative algorithm

The general framework of the TI algorithm used in this study is shown in Fig. 3. According to the problem structure of 2E-MPDP-WB, the

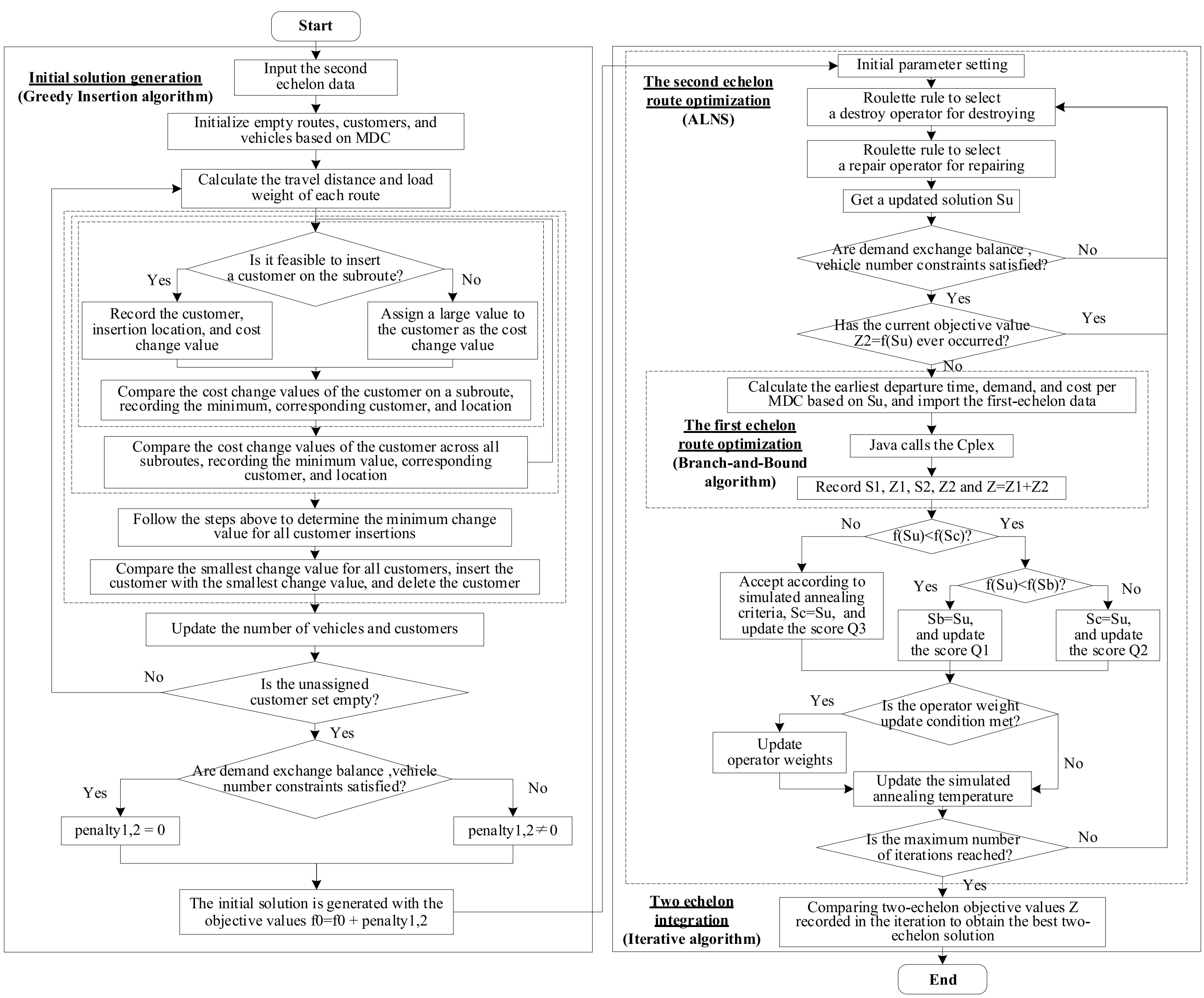

Fig. 3. General framework of the two-stage iterative algorithm.

algorithm is designed based on the two sub-models mentioned in Section 3.4. First, the initial solution of the second-echelon route is generated based on Model III using greedy insertion algorithm, and then the optimization of the second-echelon routes is completed by the implementation of the ALNS algorithm. The total operating cost of the second echelon, departure time of the earliest departing vehicle in the second-echelon vehicles of each MDC, total demand for various food materials, and respective second-echelon cost of each market are also recorded. Second, because the number of nodes involved in the first echelon is small, branch-and-bound algorithm is proposed to optimize the first-echelon sub-model based on Model IV. The results of the second-echelon sub-model is taken as input data of the first-echelon sub-model, and then CPLEX is called upon to obtain the vehicle route and total operating cost of the first echelon. Finally, the iterative idea is adopted to integrate the algorithms of two echelon sub-models. Mainly, the first-echelon calling process is embedded into the second-echelon algorithm to obtain the best solution of the two-echelon collaborative routing problem. Note that this study does not simply add the optimal costs of the first echelon and second echelon as the final result but comprehensively compares the sum of the costs of many iterations and selects the solution with the lowest cost of the two echelons as the best solution.

4.2. ALNS algorithm for the second echelon

The total demands of customers for all kinds of food materials on the second echelon determines the amount of food materials that each wholesale market on the first echelon needs to pickup and delivery. Therefore, the route optimization of the second echelon needs to be completed before the route optimization of the first echelon. This is because the second-echelon distribution problem is a MDVRP with different constraints, such as time windows, vehicle mileage, vehicle capacity, number of vehicles, and number of customers served by every depot. The number of nodes and constraints involved are more, and the model becomes more complicated. Therefore, ALNS is employed to solve the problem. ALNS, which was introduced by Ropke and Pisinger [39], is currently one of the most effective heuristic methods to solve many VRP variants [40–44].

The basic idea of ALNS is to first select a pair of destroy and repair operators according to the roulette wheel rule, and then destroy and repair the initial solution to form a new solution. Next, the operators are given different scores according to the quality of the updated solution after each iteration. When the condition of updating the operator weight is reached, the operator weights are adjusted according to the historical scores of the operators. Finally, the destroy and repair operators are selected again, and the above process is repeated until the stopping criterion is reached. As the iteration proceeds, the larger the operator weight is, the easier it is to be selected, and the effect of finding the best solution is gradually achieved.

4.2.1. Initial solution

The initial solution is the basic element to realize ALNS. The greedy insertion algorithm is used to generate the initial solution of the second-echelon distribution (Fig. 3). The main idea is to compare the changes in the cost of different insertion positions and take the position with the smallest change value to insert. The greedy insertion algorithm can be divided into five steps.

Step 1: Initialize empty routes and enter the number of customers and vehicles for each wholesale market. Initially, one empty route is set up for each wholesale market, customers are selected for insertion in the order of customer node numbers, and the cost change values of each location are compared to select the best location for insertion. Assuming that the number of customers to be inserted is n_c and the number of MDC is n_d, the number of customer nodes is 1 to n_c, and the number of MDC is (n_c+1) to (n_c+n_d).

Step 2: Judge whether it is feasible for a customer to insert a location on a sub-route, that is, whether the constraints of vehicle capacity, time window of customers, travel route length, number of vehicles, and workload balance are met. If satisfied, the customer, corresponding insertion location, and cost change value are recorded. Repeat the step, and when the node completes the insertion of all locations on that sub-route (without real insertion), compare the cost change values caused by the insertion of the node into each location on that sub-route. Additionally, record the minimum value, corresponding node, and insertion position at this location.

Step 3: Repeat step 2 to determine the minimum value of cost change for that node at each sub-route. Compare again and record the minimum value, corresponding node, and insertion position. Then the best insertion position of this node in the current solution is known.

Step 4: Repeat steps 2 and 3 to determine the minimum change values of all nodes in the current solution. Compare again and insert the node with the smallest cost change value into the corresponding position in the route. The node is deleted from the unassigned customer set.

Step 5: Repeat steps 2, 3, and 4. Stop the loop when the set of unassigned customers is empty, thus obtaining the initial solution S_0 and the objective value f_0.

Among these steps, when judging whether the workload balance and the number of vehicles are met, the cost change value is specially treated. For the workload balance constraint, if the constraint is not met, an induced value (negative value) is added to the existing cost change value; conversely, a penalty value is added. This is because the customer node insertion will give preference to the one with a small cost change value. This is done to drive customers to insert into the route with a small number of customers and ensure that the interests of each wholesale market are not reduced. The corresponding penalty function is $penalty_1 = c_1*$(existing number of customers - specified number of customers), where c_1 represents the penalty coefficient for violating the workload balance constraint. For the vehicle number constraint, if it does not meet the constraint, then a penalty function will be added on the basis of the existing cost change value; otherwise, the penalty value is 0. The corresponding penalty function is $penalty_2 = c_2*$(current number of vehicles - maximum number of vehicles), where c_2 represents the penalty coefficient for violating the vehicle number constraint. Since the penalty function is only reflected in the insertion feasibility judgment and not added to the objective value, the above two constraints are finally judged for the current initial solution. If the two constraints are not met, then the penalty value is added to the objective value; otherwise, the penalty value is set to be 0. Adding a penalty function can effectively improve the quality of the initial solution.

4.2.2. Destroy operators

The destroy operator will select a certain number of customers from the current solution to remove. At first, an empty removal set and a predetermined removal quantity $q \in [\alpha_1 n_c, \alpha_2 n_c]$ are set. The destroy operator selects customers and puts them into the removal set until the number in the set equals q. α_1 and α_2 are the removal ratios, and n_c is the total number of customers. To ensure the diversity of the search, the removal number q is randomly generated from the interval $[\alpha_1 n_c, \alpha_2 n_c]$. Seven destroy operators are used.

(1) Random removal. Randomly select the customers to be removed from the current solution and only selects one customer at a time. This process is repeated until q customer points are removed.
(2) Worst removal. The cost of each customer is equal to the objective value of the current solution minus the objective value after removing the customer. The customer with the highest cost is removed, and the above steps are repeated until q customer points are removed.

(3) Shaw removal. Similarity function can be set according to the characteristics of different problems, while it is based on distance, demand, and earliest service time in our method:

$$R_{ij} = \alpha * d_{ij} + \beta * \left|q_i - q_j\right| + \gamma * \left|e_i - e_j\right|$$

where d_{ij} is the distance between customers i and j, q_i and q_j are the demands, e_i and e_j are the earliest service times, and $\alpha+\beta+\gamma = 1$. In addition, the smaller R_{ij} is, the higher the similarity between customers i and j. First, a customer is randomly selected to be removed from the current solution, and then the similarity between each customer and the removed customer in the current solution is calculated. Take the customer with the highest similarity and remove it. Then, randomly select a customer from the removal set, continue to calculate the similarity between pairwise, and remove the customer with the highest similarity. Repeat this step until q customers are removed.

(1) Demand-based removal. Only the relevance of demand is considered, that is, $\beta = 1, \alpha = \gamma = 0$, and the rest is the same as in Shaw removal.
(2) Distance-based removal. Only the relevance of distance is considered, that is, $\alpha = 1, \beta = \gamma = 0$, and the rest is the same as in Shaw removal.
(3) Time-based removal. Only the correlation of the earliest departure time is considered, that is, $\gamma = 1, \alpha = \beta = 0$, and the rest is the same as in Shaw removal.
(4) Route removal. This operation greatly improves the diversity of algorithm search and contributes to the improvement of solution quality. Let n_c be the total number of customers and n_k be the total number of vehicles. Take n_c/n_k as the standard of route selection. Set an empty set to store sub-routes with the number of customers less than the average (n_c/n_k). If the set is not empty, randomly select a sub-route from it and remove all the customers in it; if the set is empty, randomly select one from all the sub-routes currently solved and remove all the customers in it.

4.2.3. Repair operator

The repair operator is used to reinsert the removed q customers into the previously destroyed solution one by one to form a new solution. Feasibility must be checked for reinsertion. Four repair operators are used as follows:

(1) Greedy insertion. A customer is randomly selected from the removal set, and the cost change value caused by the insertion of the customer into each position in the destroy solution is calculated. Insert the customer to the position where the cost variation value is the smallest. Repeat this operation until all q customers are inserted into the solution.
(2) Regret insertion. Let the regret value $R_e = \Delta f_1^i - \Delta f_2^i$, where Δf_1^i represents the minimum cost change value caused by the insertion of customer i, and Δf_2^i represents the second smallest cost change value caused by the insertion of customer i. Insert the customer with the largest regret value into the position with the smallest cost change value, and repeat until all q customers are inserted into the solution.
(3) Greedy noise insertion. This operation is a variant of greedy insertion. The difference is that the cost change value is added with a noise value [39] and the rest are the same as the greedy insertion.
(4) Sequential insertion. Calculate the cost change value caused by inserting a customer into all positions in the current solution and then insert the customer into the position with the smallest change value. According to the order of the customer, each customer is inserted in turn until all q customers are inserted into the solution.

After the solution of the second level has been repaired, the local search procedure 2-opt* introduced by Potvin et al. [45] is performed on all routes. The algorithm only accepts improvement solution. This procedure stops when no improving move exists in the entire neighborhood.

4.2.4. Adaptive mechanism and stopping criterion

The iterative process of ALNS can be divided into several small stages, and the operator weights are updated for each small stage iteration completed. The operator weight is calculated based on the score obtained by destroy and repair operators. The scores of a pair of destroy and repair operators reflect the historical performance of operators in the iterative process. At first, all operators have the same weight of 1 and the same score of 0. Then, according to the quality of the updated solution in each iteration, different destroy and repair operators are scored. Let $f(S_u)$ be the objective value of the updated solution, $f(S_c)$ be the objective value of the current solution, and $f(S_b)$ be the objective value of the optimal solution. When $f(S_u) < f(S_b)$, the updated solution is superior to the optimal solution, and the score of the destroy and repair operator is Q_1. When $f(S_b) < f(S_u) < f(S_c)$, the updated solution is better than the current solution but worse than the optimal solution, and the score of the destroy and repair operator is Q_2. When $f(S_u) > f(S_c)$, the updated solution is worse than the current solution. At this moment, the solution S_u is accepted according to the simulated annealing criterion, and the score of the destroy and repair operator is Q_3. The acceptance probability of simulated annealing is $e^{-(f(S_u)-f(S_c))/T}$, $T = c * T$, and $c \in (0, 1)$ is the cooling rate. The initial temperature $T_0 = \partial * f(S_0)$, $f(S_0)$is the objective value of the initial solution, and ∂ is the initial temperature control parameter. When the condition of updating the operator's weight is reached, the weight should be calculated according to the existing score: $\omega_{i,j+1} = (1-\mu)\omega_{ij} + \frac{\mu P_{ij}}{A_{ij}}$, where $\omega_{i,j+1}$ is the weight of operator i at $(j+1)$stage, P_{ij} is the sum of the scores obtained by operator i in j stage, A_{ij} is the number of times operator i was selected in j stage, and the weight adjustment coefficient $\mu \in [0, 1]$. In the previous iteration, each pair of destroy and repair operators may have an impact on the quality of the solution. However, in the later iteration process, with the historical performance of the operators, the algorithm will choose destroy and repair operators that can really improve the solution with great probability. To ensure the diversity of later solutions, we add a route removal [46] and repair before each destroy and repair. Like most literature, the stopping criterion of the ALNS algorithm in this study is also when the maximum number of iterations is reached.

4.2.5. Adjustment of departure time of vehicles on the second echelon

An inappropriate method to calculate the departure time of second-echelon vehicles will result in an error for the feasibility check. For example, we take the earliest service start time when the second-echelon vehicles arrive at the first customer minus the travel time between the market and the customer as its departure time; that is, the second-echelon vehicles do not have to wait for the time window of the first customer to open. This will lead to the time window of the second-echelon network being met, but the time relationship between the first and second echelons is not met. Therefore, we hope to reserve more route adjustment time for the first-echelon distribution by adjusting the departure time of the second-echelon vehicles to ensure the smooth operation of the TI algorithm.

The inverse solution is used to find the latest departure time of the vehicle. To determine the latest departure time of the second-echelon vehicles, we consider that the second-echelon vehicles will arrive at the last customer at its latest service starting time. The idea comes from Nagata et al. [47]. The specific calculation is shown in constraint (79), where $lat'_{k,i}$ represents the time when vehicle k arrives at customer i, and the rest of the parameters are as shown in Table 2.

$lat_{k,n} = l_n$ (*n is the last customer visited by vehicle k*)

$lat'_{k,i} = lat_{k,i+1} - t_{i,i+1} - UT \ (i = 0, \ldots, n-1)$

$$\begin{cases} lat_{k,i} = min\left\{lat'_{k,i}, l_i\right\} \ if \ lat'_{k,i} \geq e_i \\ \qquad lat_{k,i} = e_i \ if \ lat'_{k,i} < e_i \end{cases} (i = 0, \ldots, n-1) \tag{79}$$

From constraint (79), the departure time of the second-echelon vehicle $ldt_{km} = lat_{k,1} - t_{0,1}$, where $lat_{k,1}$ denotes the time of arrival of the second-echelon vehicle k at the first customer, and $t_{0,1}$ denotes the travel time of the second-echelon vehicle from MDC_m to the first customer.

4.3. Exact algorithm of the first echelon

Given that the first-echelon route involves fewer nodes, the exact algorithm (branch-and-bound algorithm) is employed to solve it. First, based on the results of the second-echelon optimization, the departure time of the earliest departing vehicle in the second-echelon vehicles of each MDC, the demand of each MDC for various food materials, and the second-echelon cost of each MDC are calculated. Moreover, the second-echelon data are imported as input parameters. Second, based on Model IV, the route optimization of the first echelon is completed using Java to call CPLEX. Finally, the optimal solution of the first echelon and the corresponding objective value are output.

Note that in the process of calling CPLEX from Java, due to the large number of iterations that need to be completed in the large-size instance, there will be problems of poor operation and stagnation. The technical reasons mainly lie in the long calling time and the accumulation of calling memory. When the second-echelon results of each iteration are different, the first-echelon problem needs to be recalculated by calling CPLEX. Some input data may lead to a long call time, so it is necessary to add a time limit for each call in the call process, which can greatly improve the call efficiency. In addition, every call will lead to memory accumulation because of the memory consumption of the exact algorithm and the large number of iterations. Therefore, every iteration needs to release the memory, which is beneficial for the whole algorithm to run quickly.

4.4. Algorithm integration for the two echelon

Different from the previous direct addition of two-echelon optimal results (hereinafter referred to as “sequential algorithm”, SeQ), this study uses an iterative idea to output the best solution of two echelons. According to the optimization results of the second echelon of each iteration, the first echelon is called once. That is, the process of the first-echelon calling is embedded into the second-echelon algorithm. However, with the increase of iteration times, a great test will consume a significant amount of running time and memory. Therefore, this study will judge whether the objective value of the second echelon occurred before each calling to avoid repeated calls.

The main idea of the two-stage algorithm integration is as follows: First, each iteration outputs a second-echelon feasible solution, and then whether the objective value corresponding to the solution appears is judged. If not, the call of the first echelon is continued; otherwise, it is not called. Second, the second-echelon solution S_2, second-echelon objective value Z_2, first-echelon objective value Z_1, and corresponding sum of the two echelon objective values Z are recorded in turn. Finally, the minimum value of Z and the corresponding first-echelon solution S_1 and second-echelon solution S_2 are compared and output. Later, in Section 5.4, the results of the sequential and TI algorithms will be compared through instances to further evaluate the effectiveness of our algorithm.

5. Computational experiments

We use computational experiments to evaluate the effectiveness of the 2E-MPDP-WB formulation and the TI algorithm. We also perform a sensitivity analysis on the application of the two-echelon collaborative logistics mode of multiple wholesale markets in an urban logistics distribution network. In Section 5.1, the generation of test instances and setting of parameters are discussed. Section 5.2 analyzes the influence of valid inequalities. Section 5.3 evaluates the effectiveness of the model and the TI algorithm. Section 5.4 carries out the sensitivity analysis of the influencing factors of the two-echelon collaborative logistics network.

The TI algorithm proposed in this study is written in Java. It runs on a Windows 10 computer configured with Intel(R) Xeon(R) W-2245 CPU @ 3.90 GHz processor with 128 GB memory using Eclipse IDE version 4.18.0 and calls ILOG CPLEX 12.6.3. When calculating the small-size instances, we use ILOG CPLEX12.6.3 to directly solve the mathematical model. CPLEX12.6.3 runs with the default settings until the optimal solution is found or the predetermined maximum calculation time is exhausted. The calculation time of all reports is in seconds. The total calculation time of CPLEX12.6.3 is limited to 5 h (18,000 s). When using the TI, each instance is repeated 10 times and the best solution is taken as the final result.

5.1. Instance generation and parameter setting

At present, there are no specific benchmark instances for the 2E-MPDP-WB model in the literature because 2E-MPDP-WB is a new variant of 2E-VRP. The nodes involved in 2E-MPDP-WB are mainly concentrated on the second-echelon distribution network, while the nodes on the first-echelon distribution are mainly related to depots. Therefore, the benchmark instances of MDVRPTW can basically meet the requirements of 2E-MPDP-WB. In the existing literature, MDVRPTW has been a classic problem studied by many scholars for many years, so there are many related benchmark instances.

First, the instances for 2E-MPDP-WB are generated based on the benchmark instances generated by Cordeau, Laporte, and Mercier [48] and specifically selected instances with customer sizes between 48 and 150 (pr01, pr02, pr03, pr07, pr08, pr11, pr12, pr13, pr17, pr18) to evaluate the quality of ALNS on the second-echelon routing problem. As the algorithm adopted by the first echelon is an exact algorithm and the ALNS algorithm is adopted by the second echelon, the uncertainty is mainly concentrated on the second echelon. The characteristics of MDVRPTW benchmark instances are shown in Table 3, including the number of depots (n_d), number of customers (n_c), number of vehicles available in each warehouse (n_k), maximum capacity of each vehicle (Q) and maximum route duration (T).

Second, 40 new instances of different sizes based on the above 10 benchmark instances are constructed to verify the effectiveness and sensitivity analysis of the TI algorithm for 2E-MPDP-WB. The information required for the new instances retains the geographic coordinates, customer demand, and customer time windows from the benchmark

Table 3
Basic information of MDVRPTW benchmark instances.

Instance	n_d	n_c	n_k	Q	T
pr01	4	48	2	200	500
pr02	4	96	3	195	480
pr03	4	144	4	190	460
pr07	6	72	2	200	500
pr08	6	144	3	190	475
pr11	4	48	1	200	500
pr12	4	96	2	195	480
pr13	4	144	3	190	460
pr17	6	72	1	200	500
pr18	6	144	2	190	475

instances; adjusts the service time for each customer, the number of vehicles owned by each wholesale market, the vehicle capacity, the maximum distance traveled by vehicles, and the number of depots and customers; and sets the number of customers initially served by each wholesale market(A_m), the maximum allowable difference(g_m), the type of food materials, and the transfer cost for each wholesale market. Each instance is represented by An_n_d_n_c or Bn_n_d_n_c, where A represents the small-size instance, B represents the large-size instance, n represents the serial number of the instance, n_d represents the number of MDCs in the instance, and n_c represents the number of all customers in the instance.

(1) Twenty small-size instances are used to compare CPLEX with the TI algorithm to verify the accuracy of the model and algorithm. Considering the complexity of our problem and the calculation ability of CPLEX12.6.3 shown by some computational experiments, the numbers of MDCs are $n_d = 2$ and 3, and the numbers of customers are n_c=15 and 30. Based on the above 10 benchmark instances, the specified number of depots and customers are randomly selected without duplication as much as possible to generate four types of small-size cases: 2 MDCs with 15 customers (A01_2_15 - A05_2_15), 3 MDCs with 15 customers (A06_3_15-A10_3_15), 2 MDCs with 30 customers (A11_2_30 - A15_2_30), and 3 MDCs with 30 customers (A16_3_30 - A20_3_30). (2) Twenty large-scale instances are used to compare the TI algorithm with the sequential algorithm to verify the effectiveness of the algorithm. Here, the numbers of MDCs are n_d=2 and 3, and the numbers of customers are n_c=48, 72, 96, and 144. Keep the original number of customers in the above 10 benchmark instances unchanged, randomly select two or three MDCs, and split one benchmark instance into two new instances. For example, pr01, pr07, pr11, and pr17 are divided into B01_2_48, B02_2_48, B07_2_72, B08_2_72 and B11_2_48, B12_2_48, B17_2_72, B18_2_72. In the same way, pr02, pr03, pr08, pr12, pr13, and pr18 are split respectively.

Other parameters in the instances are set as follows. The set of vehicles owned by each wholesale market is known, and the service time (unloading time and loading time) of each customer is randomly generated and the same in the same instance. The value of the maximum distance traveled by the vehicle is equal to the longest duration of the route in the benchmark instances. Since the order of each customer has different food materials, it is necessary to split the original demand from one type into several types. Therefore, when setting a new instance, each demand in the benchmark instances is randomly allocated to each type of food material. The first-echelon distribution is intended to transfer the demand of food materials for all customers served by each wholesale market by using vehicles with a larger capacity. The second-echelon distribution is intended to deliver the food material to each customer by using a vehicle with a relatively smaller capacity. Therefore, the capacities of the first-echelon and second-echelon vehicles are 200 and 100, respectively, and their fixed costs are 125 and 75, respectively. The travel cost per unit distance of the first- and second-echelon vehicles is set to 1, and the running speed is set to 1. The set of food material types corresponds to each wholesale market. Let the transshipment cost coefficient of each wholesale market be $L_m = 0.1(m \in V_D)$. The initial number of customers A_m in each wholesale market is randomly generated according to the wishes of the wholesale market itself. The value of maximum permissible deviation number of customers g_m is randomly selected under the condition of satisfying constraint (26), which can be used to derive $\sum_{m \in V_D} g_m \geq \sum_{m \in V_D} A_m - n_c$.

The independent distribution cost of each wholesale market can be calculated according to Model II in Section 3.2. CPLEX can be used directly to calculate the independent cost of small-size instances. When calculating the independent cost of large-size instances, the ALNS algorithm is still used to solve the problem by considering the efficiency. The specific processing method is as follows. Based on the second-echelon ALNS algorithm, the multi-depot is changed to a single depot according to Model II to adapt to VRPTW during independent distribution. The validity verification of the algorithm is shown in Section 5.3.2.

Since the initial number of customers owned by each wholesale market is different, the number of iterations for each run is set based on the total number of iterations corresponding to the new instance, proportional to the size of the number of customers. Taking A33_3_96 as an example, the total number of iterations of this instance is 20,000, and the initial number of customers of MDC1 is 57. Thus, the number of iterations of MDC1 is 11,875, that is, (57/96) $\times$ 20,000.

Finally, by keeping other parameters unchanged and changing one parameter, the parameters used by the TI algorithm in each instance are determined, as shown in Table 4. The total number of iterations is set according to the number of customers (Table 5). This is because as the number of customers increases, the performance of the algorithm improves by increasing the number of iterations within a certain range.

5.2. Influence of valid inequalities

The acceleration function of the valid inequalities (69)-(76) mentioned in Section 3.3 is first verified. The evaluation parameters include the optimal gap Gap0 and the computation time of CPU, where the optimal gap Gap0% = (UB-LB)/LB $\times$ 100, and UB and LB represent the best known solution (BKS) and best lower bound found within the time limit, respectively.

To find the best setting of valid inequalities, every possible combination of valid inequalities is used to solve small-size instances, including the setting without valid inequalities. A07_3_15 is stochastically selected as a test case. In Table 6, we test the optimal gap and computation time of each combination. We notice that among the separate use of valid inequalities (69)-(73) for the first-echelon route, the effect of inequality (69) seems to be dominant. As far as the combination with Gap0 less than 10 % is concerned, the combination containing inequalities (69)-(71) and (69)-(73) can greatly narrow the optimal gap, and the solving efficiency is comparable. However, when the three are combined, the optimal gap is 0, and the model can directly obtain the optimal solution. At the same time, the most efficient inequality combination for the first-echelon route is (69)-(73). Most of the combinations fail to obtain the optimal solution within 5 h while this combination takes only 3491 s, which is a significant improvement in computation time by more than five times. The solution efficiency is also higher when the first- and second-echelon inequalities are used together. As can be seen from Table 6, inequality (76) has the greatest influence, and the computation time is directly reduced from 3491 s to 184 s. The other combinations containing inequality (76) are also considerable and are reduced to 302, 172, and 160 s, respectively. From all combinations, the best combination of valid inequalities is (69)-(76).

The influence of the best combination of valid inequalities (69)-(76) is further analyzed on 10 small-size instances to assess their effectiveness, as shown in Table 7. In the table, the first two columns show the test instances and the current optimal value, respectively, and the last four columns show the optimal gap and computation time with or

Table 4
Parameter setting of the algorithm.

Parameter	Description	Value
Q_1, Q_2, Q_3	Scores obtained by operators at different levels	30,10,6
α, β, γ	Weight coefficient of Shaw removal operator	0.6,0.3,0.1
μ	Weight adjustment coefficient in adaptive strategy	0.1
∂	Initial temperature control parameters	0.07
c	Cooling rate in simulated annealing criterion	0.99975
q	The number of customers to be removed when using the destroy operator	$q \in [1, 0.3n]$, n is the total number of customers
c_1, c_2	Penalty coefficient for violating workload balance and vehicle number constraint	50,150

Table 5
Setting of iteration times of algorithm.

Number of customers	Total iteration times (Iterations of weight update)
$n_c = 15$	500(50)
$n_c = 30$	5000(100)
$n_c = 48$	6000(100)
$n_c = 72$	15000(100)
$n_c = 96$	20000(100)
$n_c = 144$	40000(100)

without valid inequalities, respectively. Table 7 reveals that the first five small instances can get the optimal solution within the specified time without valid inequalities, but there is still a gap between the last five small instances and the optimal solution. After the valid inequalities are added, the first five instances all obtain the optimal solution, and the computation time is reduced. Among the last five instances, two instances obtain the optimal solution and the computation time is greatly reduced compared with the ones without valid inequalities, and the other three instances obtain better solutions within the same computation time. In addition, the average increase of five instances is 19.03 %. The above results show that the proposed valid inequalities are very effective.

5.3. Performance of two-stage iterative algorithm

To evaluate the effectiveness of the proposed algorithm, 10 benchmark instances for MDVRPTW, 40 new instances (A01_2_15-A20_3_30 and B01_2_48-B20_3_144) for 2E-MPDP-WB, and 87 benchmark instances for 2E-VRP are applied in this section. The performance of ALNS on MDVRPTW, the comparison between TI algorithm and CPLEX, the comparison between TI algorithm and sequential algorithm are analyzed. Finally, TI algorithm is applied on 2E-VRP to compare our results with currently best known solutions (BSK) for each instance from the literature.

5.3.1. Performance of the two-echelon ALNS in the MDVRPTW instance

To verify the performance of the ALNS algorithm on the second-echelon routing problem, we compare the results of MDVRPTW obtained by ALNS with the BKS obtained by the most advanced algorithm in the literature [49]. The calculation results of the ALNS algorithm applied to 10 benchmark instances are shown in Table 8. The first row represents the name of the instance, and the second row indicates the BKS in the existing literature. The third and fourth rows represent the best solution in 10 runs of ALNS and the gap Gap1 from the BKS, respectively, where Gap1 % = (best solution obtained by ALNS - BKS) / BKS × 100.

From view of the average value, the results of ALNS in this study are slightly worse than the BKS by 0.31 %. From the view of the gap Gap1 with the BKS, 4 of the 10 results are the same as BKS, and the rest have the largest gap of 0.74 %. As our implementation of the ALNS algorithm is not specially designed for MDVRPTW, there is a small gap between our results and the best results in some cases. The above results show that the ALNS is highly competitive.

5.3.2. Comparison between two-stage iterative algorithm and CPLEX for 2E-MPDP-WB

Exact solutions of some small-size instances can be obtained by directly solving mathematical formulas through the exact algorithm of CPLEX12.6.3. By comparing the exact solution obtained by CPLEX with the best solution of TI, the correctness of the model and the effectiveness of the algorithm are verified. Table 9 lists the results of 20 small-size instances (A01_2_15-A20_3_30).

The first column shows the name of instances, while the second and third columns show the objective value and computation time of the exact solution obtained by CPLEX, respectively. The fourth and fifth columns show the objective value and computation time of the best solution, respectively, and the sixth column shows the percentage gap between the optimal objective value obtained by CPLEX and the objective value of the best solution obtained by TI, where Gap2 % = (Objective value of the best solution obtained by TI - optimal value obtained by CPLEX) / optimal value obtained by CPLEX × 100.

(1) From the quality of the solution, the proposed TI performs well. In seven small-size instances with exact solutions, the optimal gap Gap2 between TI and CPLEX is 0, indicating that these seven instances can be optimally solved by TI. Moreover, their computation times by TI are less than the ones by CPLEX, except for instances A03_2_15 and A07_3_15.
(2) Compared with the exact algorithm of CPLEX12.6.3, TI can obtain the best solution quickly. TI can solve each of the 20 small-size instances in a limited time. At the same time, it can be found that the instances include four groups, and the computation time between different groups is very different. The average

Table 7
Influence of the best combination of valid inequalities.

Instance	obj	no valid ineq.		(69)-(76)	
		Gap0(%)	CPU(s)	Gap0(%)	CPU(s)
A01_2_15	992.76	0.00	21.78	0.00	5.55
A02_2_15	1093.90	0.00	40.19	0.00	31.34
A03_2_15	1055.40	0.00	8.14	0.00	4.89
A04_2_15	1202.04	0.00	51.78	0.00	24.83
A05_2_15	1176.05	0.00	55.94	0.00	46.00
Avg.	1104.03	0.00	35.57	0.00	22.52
A06_3_15	1573.52	33.44	18000	20.40	18000
A07_3_15	1118.10	21.05	18000	0.00	160.03
A08_3_15	1145.09	34.82	18000	6.81	18000
A09_3_15	913.45	15.19	18000	0.00	1558.56
A10_3_15	1243.68	24.82	18000	6.94	18000
Avg.	1198.77	25.86	18000	6.83	11143.72

Table 6
Comparison of different combinations of valid inequalities.

Combination	Gap0 (%)	CPU (s)	Combination	Gap0 (%)	CPU (s)	Combination	Gap0 (%)	CPU (s)
No	21.05	18000	(71)(72)	12.64	18000	(69)(70)(71)(72)	17.09	18000
(69)	8.21	18000	(71)(73)	11.86	18000	(69)(70)(71)(73)	10.37	18000
(70)	17.93	18000	(72)(73)	13.66	18000	(69)(70)(72)(73)	9.80	18000
(71)	18.96	18000	(69)(70)(71)	11.52	18000	(69)(71)(72)(73)	7.96	18000
(72)	13.43	18000	(69)(70)(72)	12.43	18000	(70)(71)(72)(73)	10.97	18000
(73)	14.24	18000	(69)(70)(73)	10.24	18000	(69)-(73)	0.00	3491
(69)(70)	14.82	18000	(69)(71)(72)	6.36	18000	(69)-(73)(74)	0.00	15956
(69)(71)	5.24	18000	(69)(71)(73)	0.00	11241	(69)-(73)(75)	5.34	18000
(69)(72)	10.15	18000	(69)(72)(73)	8.49	18000	(69)-(73)(76)	0.00	184
(69)(73)	4.16	18000	(70)(71)(72)	24.26	18000	(69)-(73)(74)(75)	2.89	18000
(70)(71)	13.70	18000	(70)(71)(73)	11.88	18000	(69)-(73)(74)(76)	0.00	302
(70)(72)	24.01	18000	(70)(72)(73)	12.74	18000	(69)-(73)(75)(76)	0.00	172
(70)(73)	14.26	18000	(71)(72)(73)	23.40	18000	(69)-(76)	0.00	160

Table 8
Comparison of results of MDVRPTW instances.

Instance		pr01	pr02	pr03	pr07	pr08	pr11	pr12	pr13	pr17	pr18	Avg.
BKS		1074.12	1762.21	2373.65	1418.22	2096.73	1005.73	1464.50	2001.81	1236.24	1788.18	1622.14
ALNS	Best	**1074.12**	**1762.26**	**2386.38**	**1425.29**	**2101.34**	**1005.73**	**1475.39**	**2003.37**	**1236.24**	**1800.21**	**1627.03**
	Gap1(%)	0.00	0.00	0.54	0.50	0.22	0.00	0.74	0.08	0.00	0.67	0.31

Table 9
Comparison between two-stage iterative algorithm (TI) and CPLEX.

Instance	CPLEX		TI		Gap2(%)
	Obj	CPU(s)	Obj	CPU(s)	
A01_2_15	992.76	5.55	992.76	3.64	0.00
A02_2_15	1093.90	31.34	1093.90	6.11	0.00
A03_2_15	1055.40	4.89	1055.40	6.39	0.00
A04_2_15	1202.04	24.83	1202.04	3.44	0.00
A05_2_15	1176.05	46.00	1176.05	3.93	0.00
Avg.	-	22.52	-	4.70	-
A06_3_15	1573.52	18000	1573.52	57.04	0.00
A07_3_15	1118.10	160.03	1118.10	971.66	0.00
A08_3_15	1145.09	18000	1145.09	398.40	0.00
A09_3_15	913.45	1558.56	913.45	492.15	0.00
A10_3_15	1243.68	18000	1209.29	757.71	-2.77
Avg.	-	11143.72	-	535.39	-
A11_2_30	1409.93	18000	1409.93	258.47	0.00
A12_2_30	1739.86	18000	1733.29	432.51	-0.38
A13_2_30	1931.53	18000	1931.53	351.09	0.00
A14_2_30	2006.99	18000	1915.87	492.04	-4.54
A15_2_30	–	18000	2287.13	49.81	-
Avg.	-	18000	-	316.78	-
A16_3_30	–	18000	2065.41	17040.23	-
A17_3_30	1993.14	18000	1987.83	13740.07	-0.27
A18_3_30	1473.87	18000	1359.90	5100.36	-7.73
A19_3_30	–	18000	1919.75	6420.44	-
A20_3_30	1553.58	18000	1454.98	10080.69	-6.35
Avg.	-	18000	-	10476.36	-

Note: "–" means that CPLEX has not found a feasible solution within the given running time.

computation time of instances A01_2_15-A05_2_15 is only 4.7 s, while the average computation times of instances A06_3_15-A10_3_15 and A11_2_30-A15_2_30 are 535.39 and 316.78 s, respectively. However, the average computation time of instances A16_3_30-A20_3_30 is more than 2229 times that of instances A01_2_15-A05_2_15. Further analysis shows that when the number of MDCs is the same and the number of customers is different, the computation time will increase exponentially with the increase of the number of customers. When the number of customers is the same but the number of MDCs is different, even though the number of MDCs only increases by one, the speed of solving is obviously slow. The results show that the instance cannot be solved optimally by CPLEX when the number of customers is increasing, so it is more necessary to use the TI algorithm proposed in this study. The number of MDCs has a great influence on the solving efficiency of 2E-MPDP-WB.

(3) CPLEX12.6.3 requires a complete 5 h limit for 10 instances with up to 30 customers and 15 customers in 3 depots, which indicates that the solutions obtained at this time are not provable optimal solutions. In the limited time, among the 10 instances where CPLEX did not find the exact solution, the Gap2 of 6 instances was less than 0, and the Gap2 of 4 instances was 0. This result shows that TI can obtain the same or even better solution as CPLEX, and its calculation time is obviously shorter than that of CPLEX12.6.3. The other three instances cannot get the solution through CPLEX but can get a better solution through TI, which shows that the TI algorithm is better than CPLEX to some extent.

(4) The performance of CPLEX and TI algorithms is related to data characteristics. Especially, the CPLEX results of the instances A06_3_15-A10_3_15 and the TI results of the instances A16_3_30-A20_3_30 show that the computation time of the same size instances fluctuates greatly. Therefore, it can be inferred that the data characteristics of different instances will affect the performance of CPLEX and TI algorithms.

To sum up, the effectiveness of the TI algorithm is verified. Based on the correctness of the 2E-MPDP-WB mathematical model and the effectiveness of TI algorithm, TI algorithm can be used to solve large-size instances.

5.3.3. *Comparison between two-stage iterative algorithm and sequential algorithm for 2E-MPDP-WB*

To further illustrate the performance of the TI, its results are compared with the results of the sequential algorithm mentioned in Section 4.4. Table 10 gives the results of 20 large-size instances (B01_2_48-B20_3_144). The first column shows the name of the instance, the second and third columns show the objective value corresponding to the best solution obtained by TI and SeQ, respectively, and the fourth column shows the percentage gap between the objective values of TI and SeQ, where Gap3 % = (the best value obtained by TI - the best value obtained by SeQ) / the best value obtained by SeQ $\times$ 100.

As can be seen from Table 10, the TI algorithm can get the best solution for 20 large-size instances. However, when the SeQ algorithm is used, 10 instances fail to obtain the feasible solutions. The reason is that the distribution problems of two echelons are interdependent. Although the second-echelon network has obtained the best solution, the data corresponding to this solution may not meet the individual rationality requirements in the first-echelon network. This indicates that the TI can find a better feasible solution. In addition, from 10 instances where both algorithms have solutions, the objective values of TI and SQ are

Table 10
Comparison between two-stage iterative algorithm (TI) and sequential algorithm (SeQ).

Instance	SQ Obj	TI Obj	Gap3 (%)	Instance	SQ Obj	TI Obj	Gap3 (%)
B01_2_48	-	2344.65	-	B11_2_48	1977.11	1977.11	0
B02_2_48	2384.42	2151.83	-9.75	B12_2_48	-	2065.14	-
B03_3_96	3625.49	3141.55	-13.35	B13_3_96	2863.56	2863.56	0
B04_3_96	-	3429.10	-	B14_3_96	3277.01	3095.94	-5.53
B05_3_144	-	5869.76	-	B15_3_144	-	5069.12	-
B06_3_144	-	6092.35	-	B16_3_144	-	5404.50	-
B07_2_72	2781.24	2781.24	0	B17_2_72	2600.43	2600.43	0
B08_2_72	2940.07	2940.07	0	B18_2_72	-	2585.22	-
B09_3_144	-	5571.18	-	B19_3_144	-	4833.62	-
B10_3_144	5042.59	4952.90	-1.78	B20_3_144	4619.28	4619.28	0

Table 11
Basic information of 2E-VRP benchmark instances

Set	#Inst.	W_d	S_d	V_c
2a	12	1	2	21, 32
2b	9	1	2, 4	50
2c	9	1	2, 4	50
3a	12	1	2	21, 32
3b	6	1	2	50
3c	6	1	2	50
6a	27	1	4, 5, 6	50, 75, 100

consistent in 6 instances, while the objective values of TI in the other 4 instances are all lower than the objective values of SeQ, with an average improvement of about 7.60 % and a maximum improvement of 13.35 %. This finding shows that the TI algorithm can get the same or even better solution than the SeQ algorithm, that is, adding the iterative idea can improve SeQ algorithm.

5.3.4. TI algorithm applied on the 2E-VRP instance

The results of 2E-VRP obtained by TI with BKS by other meta-heuristics and exact methods are compared to further evaluate the effectiveness of TI. Considering the number of customers and satellites, we selected 7 sets of 2E-VRP benchmark instances for testing purposes. Their characteristics of are depicted in Table 11, which includes the numbers of instances of each set (#Inst.), depots (W_d), satellites (S_d), and customers (V_c). These data are available from Breunig et al. [50].

The computational results obtained by TI and gaps between BKS are provided in Table 12. In Table 12, Best is the average value of best objective values of each set obtained by TI, ABKS is the average value of BKS, and Avg. is the average gap value, where Gap4 % = (Best -ABKS)/ ABKS × 100. The BKS of each instance is summarized by Sluijk et al. [15].

Table 12 shows that TI obtains 3 ABKS out of 7 sets for 2E-VRP. The maximum gap of our best results to ABKS is 1.23 %, as set 6a shows. The average gap to ABKS for our results is 0.22 %. Although our

Table 12
Comparison of results for 2E-VRP instances.

Set		2a	2b	2c	3a	3b	3c	6a	Avg.
ABKS		577.59	549.50	607.94	591.30	714.75	668.40	910.62	
TI	Best	577.59	549.50	607.94	591.33	715.69	669.37	921.80	
	Gap4(%)	0	0	0	0.01	0.13	0.15	1.23	0.22

Table 13
Cost comparison of each MDC before and after collaboration.

Instance	MDC1			MDC2			MDC3			Gap8 (%)
	Before coll.	After coll.	Gap5 (%)	Before coll.	After coll.	Gap6 (%)	Before coll.	After coll.	Gap7 (%)	
B01_2_48	1615.35	1193.73	-26.10	1336.75	1150.92	-13.90	–	–	–	-20.58
B02_2_48	1616.74	1163.57	-38.95	1273.23	988.26	-22.38	–	–	–	-25.54
B07_2_72	2006.64	1581.89	-21.17	1539.76	1199.35	-22.11	–	–	–	-21.58
B08_2_72	2180.62	1460.31	-33.03	1522.16	1479.76	-2.79	–	–	–	-20.60
B11_2_48	1463.17	1124.54	-23.14	1176.18	852.57	-27.51	–	–	–	-25.09
B12_2_48	1666.87	1068.28	-35.91	1233.20	996.86	-19.16	–	–	–	-28.79
B17_2_72	1750.42	1248.79	-28.66	1352.48	1351.64	-0.06	–	–	–	-16.19
B18_2_72	1838.42	1498.71	-18.48	1289.90	1086.51	-15.77	–	–	–	-17.36
Avg.	-	-	-28.18	-	-	-15.46	-	-	-	-21.97
B03_3_96	2085.97	1303.37	-37.52	1547.08	1065.54	-31.13	1396.83	927.55	-33.60	-34.80
B04_3_96	2006.89	1361.80	-32.14	1428.64	1060.58	-25.76	1341.41	1006.72	-24.95	-29.49
B05_3_144	2333.81	1736.61	-25.59	2548.53	2156.00	-15.40	2492.69	1977.15	-20.68	-20.27
B06_3_144	2509.20	1862.79	-25.76	2715.66	1941.05	-28.52	2458.09	2288.69	-6.89	-27.20
B09_3_144	2142.31	1855.33	-13.40	2202.00	1949.45	-11.47	2205.47	1766.40	-19.91	-12.42
B10_3_144	2084.48	1343.44	-35.55	1984.21	1637.77	-17.46	2247.83	1971.69	-12.28	-26.73
B13_3_96	1701.26	1369.39	-19.51	1268.21	938.83	-25.97	1150.68	712.67	-38.07	-22.27
B14_3_96	1563.35	1190.89	-23.82	1265.28	966.59	-23.61	1083.66	976.10	-9.93	-23.73
B15_3_144	2097.42	1590.09	-24.19	2277.67	1763.90	-22.56	2301.43	1715.13	-25.48	-23.34
B16_3_144	1976.52	1458.60	-26.20	2240.02	1864.87	-16.75	2165.90	2081.03	-3.92	-21.18
B19_3_144	1834.27	1326.87	-27.66	2013.24	1692.04	-15.95	1992.92	1814.71	-8.94	-21.54
B20_3_144	1724.49	1195.47	-30.68	2008.78	1706.05	-15.07	1718.51	1717.76	-0.04	-22.28
Avg.	-	-	**-25.13**	-	-	**-18.61**	-	-	**-14.82**	**-21.68**

Note: "–" indicates that this instance only has two MDCs, regardless of MDC3. "Before coll." represents the cost before collaboration, while "After coll." represents the cost after collaboration.

implementation of the TI algorithm is not specially designed for 2E-VRP, these results show that TI can achieve competitive solutions. Detailed data are presented in Tables A3-A9 in the appendix, where an instance with an asterisk after BKS denotes that the best known solution of the instance is known to be optimal from Sluijk et al. [15].

5.4. Sensitivity analysis

This section will conduct a sensitivity analysis of the two-echelon collaborative routing problem from the aspects of cost comparison before and after collaboration, workload balance and related parameter change, MDC distribution, and demand mixing degree.

5.4.1. Cost comparison of each wholesale market before and after collaboration

This section applies 20 large-size instances (B01_2_48-B20_3_144) constructed in Section 5.1 to calculate and compare the cost changes of each wholesale MDC before and after collaboration.

Table 13 divides the calculation results into two types according to the number of MDCs and compares the cost of each MDC when n_d=2 and n_d=3, respectively. The first column represents the name of the instances, the second to fourth columns indicate the cost of MDC1 before collaboration (i.e., the cost of independent distribution in each wholesale market), the cost after collaboration, and the cost gap before and after collaboration (Gap5), respectively. Similarly, the last six columns represent the related results of MDC2 and MDC3, respectively, and the 11th column represents the total cost gap before and after collaboration of multiple MDCs (Gap8), where Gap5 % (Gap6 %, Gap7 %)= (cost after collaboration on one MDC - cost before collaboration on one MDC) / cost before collaboration on one MDC × 100, Gap8 % = (total cost after collaboration - total cost before collaboration) / total cost before collaboration × 100, and Avg represents the average value of the cost gap of all instances when the number of MDCs is the same.

(1) As can be seen from Table 13, regardless of whether there are two or three wholesale markets participating in the collaboration, the Gap5 of each instance is negative. Specifically, the total cost of each wholesale market after collaboration is lower than that before collaboration. This shows that for each wholesale market, collaboration is far more cost-saving than non-collaboration. The cost of each wholesale market is reduced after collaboration, and naturally the cost of the whole alliance is also greatly reduced. Compared with other studies, our model can simultaneously reduce the cost of the whole collaborative alliance and each collaborative member.
(2) It can be seen from Gap8 that the alliance has a total cost-savings of 21.97 % on average when n_d=2. The total cost of the alliance is reduced by 21.68 % on average when n_d=3. The average difference between the two types of instances is small, which shows that the number of members participating in the collaboration in this study has little influence on the overall interests.
(3) In the same instance, the costs of different MDCs are compared horizontally. For the same instance, although each MDC has achieved cost reduction after joining the collaboration, the cost-savings in each wholesale market are more or less. In some instances, the cost-savings of different MDCs before and after collaboration are relatively uniform, such as instances B07_2_72, B11_2_48, B18_2_72, B03_3_96, and B15_3_144, which achieve better fairness. There are also some instances in which there is a big gap in the cost-savings between MDCs, such as instances B17_2_72 and B20_3_144. There is one MDC that saves only a very small cost (e.g., 0.04 %) while the other MDCs save more, which may be unfair to some extent. This also indicates that work balance and fair cost allocation are not completely consistent.
(4) In different instances, the cost of each MDC is compared vertically. When n_d=2, MDC1 saves 38.95 % at the maximum, 18.48 % at the minimum, and 28.18 % on average in 8 instances. MDC2 saves 27.51 % at the maximum, 0.06 % at the minimum, and 15.46 % on average. When n_d=3, MDC1 saves 37.52 % at the maximum, 13.40 % at the minimum, and 25.13 % on average in 12 instances. MDC2 saves 31.13 % at the maximum, 11.47 % at the minimum, and 18.61 % on average. MDC3 saves 38.07 % at maximum, 0.04 % at minimum, and 14.82 % on average. It can be seen that in the collaborative alliance that while there are great and small advantages for some MDCs, each MDC can get at least 14.82 % cost-savings on average.

5.4.2. Analysis of workload balance

Workload balance is an important factor to ensure the fairness of allocation of customers in each wholesale market in the process of collaboration. In the process of collaboration in multiple wholesale markets, the alliance often overlooks the characteristics and needs of each partner because it excessively pursues the maximization of the overall interests, such as the inability to ensure the corresponding customer source. Therefore, to study the impact of workload balance on the total cost, the results are compared with those after the constraint on existing small-size instances is deleted, as shown in Table 14. In the table, the first and fifth columns represent the name of the instances, and the second and third columns represent the total cost with and without workload balance, respectively. The sixth and seventh columns are the same as the second and third columns. Gap 9 % = (total cost with balance - total cost without balance) / total cost without balance × 100, and Avg represents the average value of the cost gap of all instances when the number of MDCs and customers are the same.

Table 14 illustrates that Gap9 is greater than or equal to 0, that is, the total cost with workload balance constraint is higher than that without it. This shows that all wholesale markets need to pay a certain price if they consider the workload balance when collaborating.

In addition, it can be found that when the size of the alliance or the number of customers owned by each alliance member is different, the impact of having or having no workload balance on the total cost of the entire alliance is also different. The Gap 9 of two MDCs is 1.46 % and

Table 14
Cost comparison of the total cost with and without workload balance constraint.

Instance	Balance	No balance	Gap9(%)	Instance	Balance	No balance	Gap9(%)
A01_2_15	992.76	992.76	0.00	A11_2_30	1409.93	1409.93	0.00
A02_2_15	1093.9	1071.07	2.13	A12_2_30	1733.29	1710.4	1.34
A03_2_15	1055.4	1048.57	0.65	A13_2_30	1931.53	1893.45	2.01
A04_2_15	1202.04	1169.09	2.82	A14_2_30	1915.87	1911.67	0.22
A05_2_15	1176.05	1156.27	1.71	A15_2_30	2287.13	2196.37	4.13
Avg.	-	-	**1.46**	Avg.	-	-	**1.54**
A06_3_15	1573.52	1519.86	3.53	A16_3_30	2065.41	2002.56	3.14
A07_3_15	1118.1	1095.83	2.03	A17_3_30	1987.83	1830.39	8.60
A08_3_15	1145.09	1091.91	4.87	A18_3_30	1359.9	1283.01	5.99
A09_3_15	913.45	876.69	4.19	A19_3_30	1919.75	1883.65	1.92
A10_3_15	1209.29	1179.81	2.50	A20_3_30	1454.98	1430.01	1.75
Avg.	-	-	**3.42**	Avg.	-	-	**4.28**

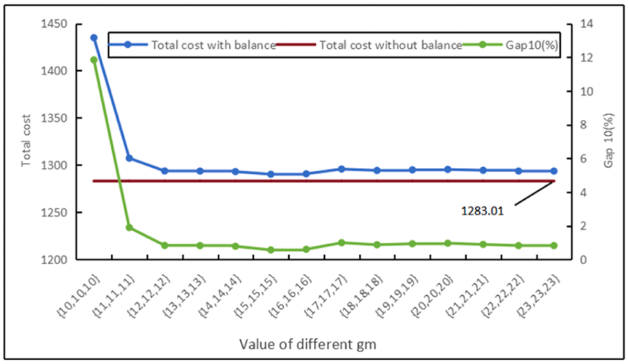

Fig. 4. Trend of total cost changes under different g_m.

1.54 %, respectively, and that of three MDCs is 3.42 % and 4.28 %, respectively. Clearly, the cost of considering workload balance when there are three MDCs as alliance members is significantly higher than when there are only two MDCs. This shows that the more wholesale markets participate in collaboration, the more difficult it is to implement workload balance. At the same time, when the number of MDCs is the same, the larger the number of customers, the higher the cost of implementing this balance. Thus, changes in the number of wholesale markets have a greater impact on total costs than changes in the number of customers.

From the above research, the impact of workload balance on different alliances is either large or small. The workload balance constraint mainly consists of the initial number of customers A_m and the maximum allowable deviation of customers g_m. To further study how the workload balance affects the alliance, it will be discussed from the following aspects:

(1) Impact of workload balance on total cost under different g_m

Select the instance A18_3_30 with a bigger gap value (Gap 9 = 5.99 %) from Table 14 to further study the impact of the maximum allowable difference g_m in each wholesale market. Keep other parameters unchanged and only change the value of g_m. Where the initial number of customers A_m={20,17,23}, when the constraint (26) is satisfied, g_m is set to {10,10,10}, {11,11,11},..., {23,23,23}. Fig. 4 shows the total cost with and without workload balance, where Gap10 % = (total cost with balance - total cost without balance) / total cost without balance × 100, and the specific data are shown in Table A1 in the Appendix.

As can be seen from Fig. 4, with the gradual increase of g_m, the total cost and Gap 10 after adding the workload balance show a trend of the decline first and then flatten out. When g_m={10,10,10}, Gap 10 is the largest, and after g_m={12,12,12}, Gap 10 remains stable at about 0.80 %. This outcome indicates that it takes great cost to take the minimum maximum allowable deviation customers g_m when participants implement the workload balance. That is, participants have a little more tolerance for the deviation in the number of customers after collaboration, and a little more relaxation of the requirements can save a lot of costs. Therefore, when implementing workload balance, alliance participants can slightly lower their requirements for customer number deviation, which can avoid high costs as well as play a balancing role.

(2) Impact of workload balance on customer number allocation under different A_m

To explore the impact of workload balance on the customer number allocation, take A18_3_30 as an example, change the value of A_m, and compare customer number allocation with and without workload balance after collaboration. The detailed results are shown in Table 15. For the same instance, the g_m of each wholesale market is set to the same and meets constraint (26). Table 15 includes four groups. "Before coll." represents the comparison of the initial customers' numbers before collaboration. "Balance" represents the change in customer number allocation after adding the workload balance constraint. "No balance" indicates the change of customer number allocation when there is no workload balance constraint. The fourth group shows the fairness judgment of the customer number allocation, namely, judging the presence or absence of (a) a higher number of initial customers owned by one wholesale market before collaboration but still took on more customers after collaboration or (b) fewer initial customers owned by one wholesale market before collaboration but still committed to fewer customers after collaboration. If all members of an alliance meet this rule, it indicates that the customer number allocation in this case is fair. Among them, ①, ②, and ③ represent MDC1, MDC2, and MDC3, respectively. Prop. represents the proportion of customers owned by MDC1, MDC2, and MDC3, respectively, to the total number of customers under different circumstances. Prop. comparison represents the comparison of the proportion of customers owned by each wholesale market in different cases. A fair choice indicates which is fairer with or without workload balance in terms of the customer number allocation after collaboration.

Take A18_3_30 as an example where Am={26,18,10}, the proportions of the initial number of customers owned by MDC1, MDC2, and MDC3 to the total number of customers are 48 %, 33 %, and 19 %, respectively. At this time, the proportion of the number of customers owned by each wholesale market is MDC1>MDC2>MDC3. After workload balance is added to collaboration, the proportions of customers allocated to three wholesale markets are 57 %, 30 %, and 13 %, respectively. It is still MDC1>MDC2>MDC3, which meets the fairness requirements. When the workload balance is not added in the collaboration, the proportions of customers allocated by three wholesale markets are 63 %, 17 %, and 20 %, respectively, that is, MDC1>MDC3>MDC2. A deviation can be found from the proportion of the initial number of customers in each wholesale market before collaboration. Originally, the number of customers owned by MDC2 ranked second but then ranked third after collaboration, resulting in the

Table 16
Impact of expanding the time window on the total cost.

Expansion multiple of the time window	1x	2x	3x	4x	5x
Total cost with balance	1987.83	1598.51	1563.17	1554.03	1528.66
Total cost without balance	1830.39	1519.63	1496.56	1496.56	1496.56
Gap11(%)	8.60	5.19	4.45	3.84	2.14

Table 15
Impact of workload balance on the allocation of customers under different initial number of customers A_m.

Before coll.			Balance		No balance		The fairness judgment of the customer number allocation		
Am	Prop. (%)	Prop. comparison	Customer allocation	Prop.1 (%)	Customer allocation	Prop.2 (%)	Prop.1 comparison	Prop.2 comparison	A fair choice
{26,18,10}	48,33,19	①>②>③	{17,9,4}	57,30,13	{19,5,6}	63,17,20	①>②>③	①>③>②	Balance
{26,19,12}	46,33,21	①>②>③	{17,10,3}	57,33,10	{19,5,6}	63,17,20	①>②>③	①>③>②	Balance
{26,20,14}	43,33,23	①>②>③	{17,9,4}	57,30,13	{12,12,6}	40,40,20	①>②>③	①=②>③	Balance
{23,18,13}	43,33,24	①>②>③	{17,9,4}	57,30,13	{19,5,6}	63,17,20	①>②>③	①>③>②	Balance
{28,24,20}	39,33,28	①>②>③	{14,10,6}	47,33,20	{18,6,6}	60,20,20	①>②>③	①>②=③	Balance

situation where the wholesale market that originally undertook a large number of customers undertook a small number of customers. Similarly, when observing other situations, we find that the number of customers realized by adding workload balance in the collaboration process is more equitable. This shows that it is necessary for different enterprises to implement workload balance in the process of collaboration.

(3) Impact of workload balance on total cost under different customer time windows

Customer time window is an important factor affecting the total cost of the distribution. To meet the time window requirements of customers, enterprises sometimes send a separate vehicle for distribution, which increases the additional cost. Take A17_3_30 with a maximum gap value as an example. Keep other parameters unchanged, and enlarge the time window to further explore the impact of workload balance on the total cost under different customer time windows. The detailed results are shown in Table 14, where Gap11 % = (total cost with balance - total cost without balance) / total cost without balance × 100. Table 16 shows that as the time window of customers multiplies, Gap11 gradually decreases. Specifically, the cost gap with or without workload balance is gradually narrowed, which indicates that the relaxation of the customer time window is very beneficial to the implementation of workload balance in the collaboration.

5.4.3. Influence of MDC distribution on collaboration cost

To study how MDC distribution affects the collaboration cost, we keep the location of customers unchanged (the distribution of customers is relatively uniform at this time) and scale the MDC coordinates of instance B14_3_96 in equal proportion to generate the MDC distribution in 10 situations, as shown in Fig. 5. In the figure, the red triangle area represents the area surrounded by three MDCs under different situations, which correspond to situations 1–10 from the inside to outside, respectively. For example, situation 1 represents the innermost red triangle distribution, that is, three MDCs are located at points A, B, and C, respectively. Through calculation, situations 1–9 are found to have solutions, but there are no solutions after situation 10. This shows that when an MDC is too far away from the other MDCs, collaboration between wholesale markets cannot be formed or collaboration breaks down. Therefore, Fig. 5 only shows the results of situations 1–9.

As can be seen from Fig. 6, with the gradual expansion of the distance between MDCs, the overall change trend of the total cost is to decrease first and then increase, in which the lowest total cost is 3,016.87 in situation 3 and the highest is 3,835.53 in situation 9. This result shows that when three wholesale markets collaborate, it is not appropriate for MDCs to be too close or too far apart, and there may be an optimal MDC distribution for the current customer distribution. Therefore, through this study, an enterprise can choose the best partner according to its MDC location to promote the most favorable collaboration. In addition, it can be found that the cost change between situations 1–7 is small, while the cost change after situation 7 is large, even directly leading to the breakdown of collaboration. This indicates that building an MDC in the area around the optimal MDC distribution has a smaller impact on the total cost of the alliance, while the greater the distances among the MDCs are, the more significant the increase in the total cost of the alliance is. Therefore, when building a new MDC or choosing a wholesale market that they want to collaborate with, enterprises should try to choose the closer wholesale market to cooperate with.

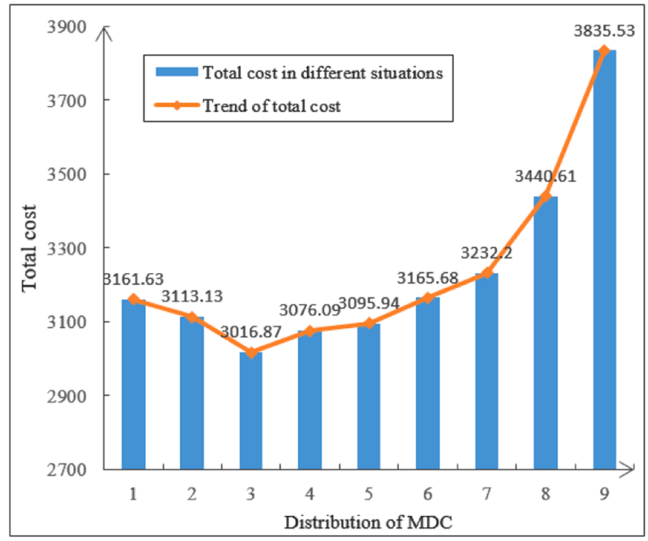

Fig. 6. Total cost changes under different MDC distributions.

5.4.4. Analysis of demand mixing degree

The demand of each customer for various food materials is an important factor in determining the benefits of the alliance. In real life, each customer buys the food materials needed for the day according to the actual usage of the previous day, and so the demand is different. Some customers may only need to go to the vegetable wholesale market to buy fruits and vegetables, while some may have a demand for vegetables, meat, poultry, and aquatic products. Given that this study stipulates that a wholesale market only represents one kind of food material, if customers do not need certain food materials, vehicles do not need to pass through the wholesale market. The distribution route will change as well, and the total cost of the alliance will correspondingly be affected. Therefore, we further discuss the impact on the total cost of the alliance

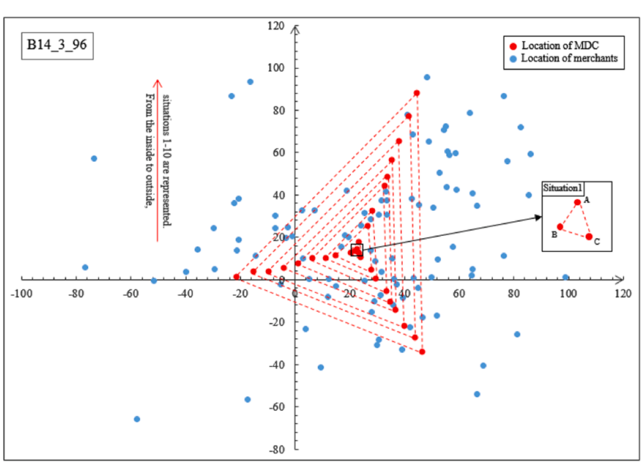

Fig. 5. Distribution of MDCs in different situations (taking A34_3_96 as an example).

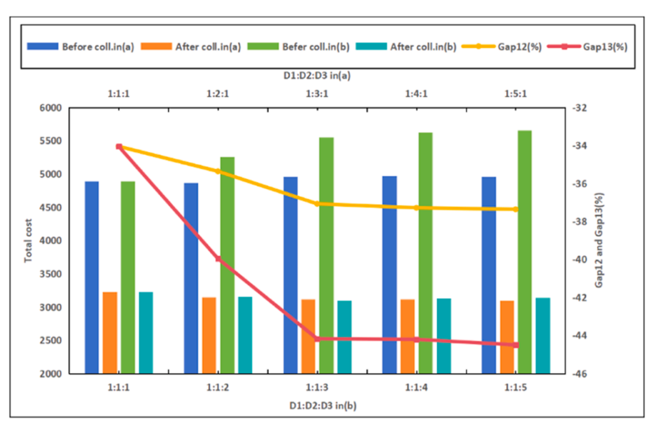

Fig. 7. Comparison of total cost under different requirements mixing degrees. "Before coll." represents the cost before collaboration, while "After coll." represents the cost after collaboration.

based on the mixed demand degree of all customers for food materials.

We take the instance B14_3_96 as an example and consider three situations: customers only need one kind of food material (D1), customers need two kinds of food materials (D2), and customers need all three kinds of food materials (D3). We mainly discuss the change of the total cost of the alliance when the proportion of customers D2 or D3 increases gradually. Since the demand for only one kind of food material is equivalent to the special case of the demand for two or three kinds of food materials, we can infer the law of the former by studying the law of the latter two. Fig. 7 shows the change in total cost under different mixing ratios with gradually increasing proportions of customers D2 or D3. Case (a) and case (b) indicate the cases of the two and three kinds of food materials with increasing proportion, respectively. The mixing ratio is expressed as D1:D2:D3. Gap12 % (Gap13 %) = (total cost after collaboration - total cost before collaboration) / total cost before collaboration × 100. Detailed data are presented in Table A2 in the appendix.

Table A2 and Fig. 7 indicate that both Gap12 and Gap13 show a decreasing trend, that is, the cost gap gradually increases. This finding shows that the higher the proportion of customers D2 or D3, the more the total cost of the alliance will be saved. In addition, by observing the average value of Gap12and Gap13, it can be found that with the increase of the same proportion, as the proportion of D2 gradually increases, the total cost is saved by 36.22 % on average while the average savings with the same increase of D3 is 41.37 %. This shows that under the same proportion, the number of D3 accounts for a high proportion, which is better than the collaboration effect of D2 with the same proportion. That is, the higher the proportion of customers who have a demand for various food materials, the more obvious the advantages of collaboration among multiple wholesale markets. Therefore, before participating in the collaboration, the enterprise can evaluate the customers that it serves itself or by other members from the perspective of demand mixing degree (i.e., the proportion of customers who need a variety of food materials) to see whether it is suitable to join the alliance or choose this enterprise as a partner.

6. Conclusions

In this study, 2E-MPDP-WB is introduced by considering the different types of food materials in different wholesale markets and the needs of customers with various food materials. 2E-MPDP-WB involves the route optimization of two echelons. The first echelon carries out the exchange of food materials between different wholesale markets, which is equivalent to the PDPMDTD. The second echelon distributes the food materials gathered in each wholesale market to the customers, which is equivalent to the MDVRPTW. First, we establish a formulation for 2E-MPDP-WB, with the objective of minimizing the total cost. Among them, route optimization and benefit balance are integrated into one, that is, the cost allocation of partners is completed at the same time of route optimization. Second, a TI algorithm is proposed to solve 2E-MPDP-WB. The 2E-MPDP-WB model is decomposed into two sub-models, which are solved by branch-and-bound algorithms and the ALNS, respectively. Then the first-echelon calling process is embedded into the second-echelon algorithm through an iterative idea to realize the integration of the two echelon algorithms. Finally, the correctness of the 2E-MPDP-WB model and the effectiveness of the TI algorithm are evaluated through computational experiments, and sensitivity analysis are conducted from four aspects: cost comparison before and after collaboration, workload balance and related parameter change, MDC distribution, and demand mixing degree. The following conclusions and management insights are drawn:

(1) Collaborative distribution in multiple wholesale markets is beneficial. It can not only reduce the total cost of the alliance but also realize the cost-savings of each partner while optimizing the route by workload balance and individual rationality constraints, which is conducive to ensuring the partners' own interests.
(2) Implementing workload balance among multiple wholesale markets during collaboration can help achieve fairness in customer allocation and long-term stability of the alliance. However, it can likewise slightly increase the cost of collaboration, especially when there are more wholesale markets participating in collaboration or the number of customers is higher. Cost added also depends on the allowable customer number deviation value and the width of the time window. Therefore, the enterprises can reduce the cost increase when implementing the workload balance by slightly reducing the requirements of the allowable customer number deviation or appropriately relaxing the time windows of customers.
(3) An optimal MDC distribution for a given customer distribution may exist. Therefore, for an enterprise, it can choose the best partner according to the location of its distribution center and promote the most favorable collaboration. Furthermore, when building a new MDC or choosing a company to collaborate with, enterprises should try to choose the closer wholesale market.
(4) Enterprises can evaluate the demand mixing degree (i.e., the proportion of customers who need a variety of food materials) when considering whether they can join the alliance or choose which partner. The higher the proportion of customers who need a variety of food materials, the more obvious the advantages of collaboration will be.

In future research, there are two directions to be further explored. First, considering factors such as carbon emissions and customer satisfaction, establish a multi-objective two-echelon model at the alliance level. Second, when establishing a multi-objective model combining the goals of alliance and fairness of individual partners for each partner, the cost-savings achieved are more or less.

CRediT authorship contribution statement

Jian Li: Writing – original draft, Methodology, Funding acquisition, Conceptualization. **Lu Cang:** Formal analysis, Writing – original draft, Software, Methodology. **Yisheng Wu:** Formal analysis. **Zhaotong Zhang:** Supervision, Investigation, Formal analysis.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgment

The authors gratefully acknowledge valuable comments of two anonymous referees, which considerably improved the quality of our work. This work was supported by the National Key Research and Development Program of China (2018YFB1403100), the Humanities and Social Sciences Youth Foundation of Ministry of Education of China (19YJC630079), Jiangsu Provincial Social Science Fund (22GLD006), and Humanities and Social Sciences Research Fund for Fundamental Research of Central Universities of Nanjing Agricultural University (SKYZ2023042).

Appendix A

Table A1
Change of total cost under different g_m

	Balance	No balance	Gap10(%)
{10,10,10}	1434.88	1283.01	11.84
{11,11,11}	1307.25	1283.01	1.89
{12,12,12}	1293.70	1283.01	0.83
{13,13,13}	1293.59	1283.01	0.82
{14,14,14}	1293.03	1283.01	0.78
{15,15,15}	1290.21	1283.01	0.56
{16,16,16}	1290.70	1283.01	0.60
{17,17,17}	1295.72	1283.01	0.99
{18,18,18}	1294.21	1283.01	0.87
{19,19,19}	1294.88	1283.01	0.93
{20,20,20}	1295.30	1283.01	0.96
{21,21,21}	1294.40	1283.01	0.89
{22,22,22}	1293.59	1283.01	0.82
{23,23,23}	1293.59	1283.01	0.82

Table A2
The change of total cost under different demand mixing degrees.

cc	Before coll. in(a)	After coll. in(a)	Gap12 (%)	Mixing proportion(b)	Before coll. in(b)	After coll. in(b)	Gap13 (%)
1:1:1	4896.98	3229.50	-34.05	1:1:1	4896.98	3229.50	-34.05
1:2:1	4870.18	3148.15	-35.36	1:1:2	5260.00	3158.46	-39.95
1:3:1	4958.77	3120.62	-37.07	1:1:3	5554.14	3100.94	-44.17
1:4:1	4972.26	3118.45	-37.28	1:1:4	5622.29	3137.31	-44.20
1:5:1	4955.24	3103.99	-37.36	1:1:5	5656.63	3139.95	-44.49
Avg.	-	-	-36.22	Avg.	-		-41.37

Note: The mixing proportion (a) and (b) indicate the mixing proportion of the customers with the two and three kinds of food materials with increasing proportion. Avg represents the average value of the cost gap.

Table A3
Results for Set 2a instances of 2E-VRP.

Instance	V_c	S_d	K_1	K_2	TI			BKS	Gap4_1(%)
					Avg	Best	Time(s)		
E-n22-k4-s6-17	21	2	3	4	417.07	417.07	6	417.07*	0.00 %
E-n22-k4-s8-14	21	2	3	4	384.96	384.96	8	384.96*	0.00 %
E-n22-k4-s9-19	21	2	3	4	470.60	470.60	7	470.60*	0.00 %
E-n22-k4-s10-14	21	2	3	4	371.50	371.50	7	371.50*	0.00 %
E-n22-k4-s11-12	21	2	3	4	427.22	427.22	7	427.22*	0.00 %
E-n22-k4-s12-16	21	2	3	4	392.78	392.78	6	392.78*	0.00 %
E-n33-k4-s14-22	32	2	3	4	779.05	779.05	15	779.05*	0.00 %
E-n33-k4-s1-9	32	2	3	4	730.16	730.16	14	730.16*	0.00 %
E-n33-k4-s2-13	32	2	3	4	714.63	714.63	15	714.63*	0.00 %
E-n33-k4-s3-17	32	2	3	4	707.48	707.48	15	707.48*	0.00 %
E-n33-k4-s4-5	32	2	3	4	778.74	778.74	17	778.74*	0.00 %
E-n33-k4-s7-25	32	2	3	4	756.84	756.84	15	756.85*	0.00 %
Avg.						577.59	11	577.59	0.00 %

Table A4
Results for Set 2b instances of 2E-VRP.

Instance	V_c	S_d	K_1	K_2	TI			BKS	Gap4_1(%)
					Avg	Best	Time(s)		
E-n51-k5-s11-19	50	2	3	5	584.83	581.64	36	581.64*	0.00 %
E-n51-k5-s11-19-27-47	50	4	4	5	527.63	527.63	43	527.63*	0.00 %
E-n51-k5-s2-17	50	2	3	5		597.49	41	597.49*	0.00 %
E-n51-k5-s2-4-17-46	50	4	4	5	530.76	530.76	33	530.76*	0.00 %
E-n51-k5-s27-47	50	2	3	5		538.22	42	538.22*	0.00 %

(continued on next page)

Table A4 (*continued*)

Instance	V_c	S_d	K_1	K_2	TI			BKS	Gap4_1(%)
					Avg	Best	Time(s)		
E-n51-k5-s32-37	50	2	3	5		552.28	36	552.28*	0.00 %
E-n51-k5-s4-46	50	2	3	5	530.76	530.76	31	530.76*	0.00 %
E-n51-k5-s6-12	50	2	3	5	554.81	554.81	43	554.81*	0.00 %
E-n51-k5-s6-12-32-37	50	4	4	5	531.92	531.92	47	531.92*	0.00 %
Avg.						549.50	39	549.50	0.00 %

Table A5
Results for Set 2c instances of 2E-VRP.

Instance	V_c	S_d	K_1	K_2	TI			BKS	Gap4_1(%)
					Avg	Best	Time(s)		
E-n51-k5-s11-19	50	2	3	5	617.41	617.41	31	617.42	0.00 %
E-n51-k5-s11-19-27-47	50	4	4	5	538.13	530.76	40	530.76	0.00 %
E-n51-k5-s2-17	50	2	3	5	601.44	601.39	39	601.39	0.00 %
E-n51-k5-s2-4-17-46	50	4	4	5	601.55	601.39	33	601.39	0.00 %
E-n51-k5-s27-47	50	2	3	5	530.76	530.76	29	530.76	0.00 %
E-n51-k5-s32-37	50	2	3	5	754.26	752.59	35	752.59	0.00 %
E-n51-k5-s4-46	50	2	3	5	702.33	702.33	40	702.33	0.00 %
E-n51-k5-s6-12	50	2	3	5	567.42	567.42	41	567.42	0.00 %
E-n51-k5-s6-12-32-37	50	4	4	5	568.67	567.42	34	567.42	0.00 %
Avg.						607.94	36	607.94	0.00 %

Table A6
Results for Set 3a instances of 2E-VRP.

Instance	V_c	S_d	K_1	K_2	TI			BKS	Gap4_2(%)
					Avg	Best	Time(s)		
E-n22-k4-s13-14	21	2	3	4	526.15	526.15	7	526.15*	0.00 %
E-n22-k4-s13-16	21	2	3	4	521.09	521.09	8	521.09*	0.00 %
E-n22-k4-s13-17	21	2	3	4	496.38	496.38	7	496.38*	0.00 %
E-n22-k4-s14-19	21	2	3	4	498.80	498.80	7	498.80*	0.00 %
E-n22-k4-s17-19	21	2	3	4	512.80	512.80	6	512.80*	0.00 %
E-n22-k4-s19-21	21	2	3	4	520.42	520.42	8	520.42*	0.00 %
E-n33-k4-s16-22	32	2	3	4	672.68	672.17	16	672.17*	0.00 %
E-n33-k4-s16-24	32	2	3	4	666.06	666.02	15	666.02*	0.00 %
E-n33-k4-s19-26	32	2	3	4	680.77	680.77	15	680.36*	0.01 %
E-n33-k4-s22-26	32	2	3	4	680.79	680.36	14	680.36*	0.00 %
E-n33-k4-s24-28	32	2	3	4	670.42	670.42	16	670.43*	0.00 %
E-n33-k4-s25-28	32	2	3	4	650.56	650.54	17	650.58*	-0.01 %
Avg.						591.33	11	591.30	0.01 %

Table A7
Results for Set 3b instances of 2E-VRP.

Instance	V_c	S_d	K_1	K_2	TI			BKS	Gap4_2(%)
					Avg	Best	Time(s)		
E-n51-k5-s12-18	50	2	3	5	692.39	690.59	35	690.59	0.00 %
E-n51-k5-s12-41	50	2	3	5	685.76	683.05	31	683.05	0.00 %
E-n51-k5-s12-43	50	2	3	5	712.10	710.41	36	710.41	0.00 %
E-n51-k5-s39-41	50	2	3	5	730.15	729.45	34	728.54	0.12 %
E-n51-k5-s40-41	50	2	3	5	729.46	728.54	32	723.75	0.66 %
E-n51-k5-s40-43	50	2	3	5	753.27	752.12	36	752.15	0.00 %
Avg.						715.69	34	714.75	0.13 %

Table A8
Results for Set 3c instances of 2E-VRP.

Instance	V_c	S_d	K_1	K_2	TI			BKS	Gap4_2(%)
					Avg	Best	Time(s)		
E-n51-k5-s13-19	50	2	3	5	561.01	560.73	36	560.73*	0.00 %
E-n51-k5-s13-42	50	2	3	5	565.09	564.45	33	564.45*	0.00 %
E-n51-k5-s13-44	50	2	3	5	565.55	564.45	33	564.45*	0.00 %
E-n51-k5-s40-42	50	2	3	5	748.96	748.39	31	746.31*	0.28 %
E-n51-k5-s41-42	50	2	3	5	775.51	771.56	32	771.56*	0.00 %
E-n51-k5-s41-44	50	2	3	5	807.06	806.63	33	802.91*	0.46 %
Avg.						669.37	33	668.40	0.15 %

Table A9
Results for Set 6a instances of 2E-VRP.

Instance	V_c	S_d	K_1	K_2	TI			BKS	Gap4_3(%)
					Avg	Best	Time(s)		
A-n51-4	50	4	2	50	656.73	656.50	46	652.00*	0.69 %
A-n51-5	50	5	2	50	665.49	663.41	46	663.41*	0.00 %
A-n51-6	50	6	2	50	666.11	665.93	57	662.51*	0.51 %
A-n76-4	75	4	3	75	1008.09	1007.66	152	985.95*	2.15 %
A-n76-5	75	5	3	75	987.52	987.35	178	979.15*	0.83 %
A-n76-6	75	6	3	75	993.71	985.12	177	970.20*	1.51 %
A-n101-4	100	4	4	100	1199.94	1195.87	400	1194.17*	0.14 %
A-n101-5	100	5	4	100	1236.41	1231.27	413	1211.38*	1.62 %
A-n101-6	100	6	4	100	1162.19	1149.51	422	1155.96	-0.56 %
B-n51-4	50	4	2	50	567.64	563.98	40	563.98*	0.00 %
B-n51-5	50	5	2	50	551.06	549.22	59	549.23*	0.00 %
B-n51-6	50	6	2	50	565.74	564.71	44	556.32*	1.49 %
B-n76-4	75	4	3	75	809.88	806.53	159	792.73*	1.71 %
B-n76-5	75	5	3	75	786.78	785.91	181	783.93*	0.21 %
B-n76-6	75	6	3	75	786.63	784.78	195	774.17*	1.49 %
B-n101-4	100	4	4	100	950.68	946.76	353	939.21*	0.80 %
B-n101-5	100	5	4	100	980.77	978.43	401	967.82*	1.08 %
B-n101-6	100	6	4	100	965.27	964.32	404	960.29*	0.42 %
C-n51-4	50	4	2	50	689.69	689.18	39	689.18*	0.00 %
C-n51-5	50	5	2	50	744.39	742.66	46	723.12*	2.63 %
C-n51-6	50	6	2	50	700.98	699.27	58	697.00*	0.32 %
C-n76-4	75	4	3	75	1067.18	1063.08	148	1054.89*	0.77 %
C-n76-5	75	5	3	75	1135.49	1133.17	175	1115.32*	1.57 %
C-n76-6	75	6	3	75	1087.14	1084.43	176	1060.52	2.20 %
C-n101-4	100	4	4	100	1312.64	1309.71	370	1302.16	0.58 %
C-n101-5	100	5	4	100	1349.96	1345.68	414	1305.82	2.96 %
C-n101-6	100	6	4	100	1336.19	1334.05	372	1284.48	3.72 %
Avg.						921.80	205	910.62	1.23 %

References

[1] Guajardo M, Rönnqvist M. A review on cost allocation methods in collaborative transportation. Int. Transact. Oper. Res. 2016;23:371–92.

[2] Matl P, Hartl R, Vidal T. Workload equity in vehicle routing: The impact of alternative workload resources. Comput. Oper. Res. 2019;110:116–29.

[3] Mancini S, Gansterer M, Hartl R. The collaborative consistent vehicle routing problem with workload balance. Eur. J. Oper. Res. 2021;293(3):955–65.

[4] Hezarkhani B, Slikker M, Van Woensel T. Gain-sharing in urban consolidation centers. European Journal of Operational Research 2019;279(2):380–92.

[5] Defryn C, Sörensen K, Dullaert W. Integrating partner objectives in horizontal logistics optimisation models. Omega (Westport) 2019;82:1–12.

[6] Vanovermeire C, Sörensen K. Integration of the cost allocation in the optimization of collaborative bundling. Transport. Res. Part E Logist. Transport. Rev. 2014;72: 125–43.

[7] Liu L, Liao W. Optimization and profit distribution in a two-echelon collaborative waste collection routing problem from economic and environmental perspective. Waste Manage. 2021;120:400–14.

[8] Wang Y, Ma XL, Liu MW, Gong K, Liu Y, Xu MZ, Wang YH. Cooperation and profit allocation in two-echelon logistics joint distribution network optimization. Appl. Soft. Comput. 2017;56:143–57.

[9] Rabbouch B, Mraihi R, Saadaoui F. A recent brief survey for the multi depot heterogenous vehicle routing problem with time windows. Int. Conf. Hybrid Intell. Syst. 2018;734:147–57.

[10] Wang Y, Peng S, Zhou S, Mahmoudi M, Zhen L. Green logistics location-routing problem with eco-packages. Transport. Res. Part E Logist. Transport. Rev. 2020; 2020(143):102118.

[11] Wang Y, Wang X, Wei Y, Sun Y, Fan J, Wang H. Two-echelon multi-depot multi-period location-routing problem with pickup and delivery. Comput. Ind. Eng. 2023;2023(182):109385.

[12] Koç Ç, Laporte G, Tükenmez İ. A review of vehicle routing with simultaneous pickup and delivery. Comput. Oper. Res. 2020;122:104987.

[13] Crainic TG, Ricciardi N, Storchi G, Storchi G. Models for evaluating and planning city logistics systems. Transport. Sci. 2009;43(4):432–54.

[14] Perboli G, Tadei R, Vigo D. The two-echelon capacitated vehicle routing problem: models and math-based heuristics. Transport. Sci. 2011;45(3):364–80.

[15] Sluijk N, Florio AM, Kinable J, Dellaert N, Woense T. Two-echelon vehicle routing problems: A literature review. Eur. J. Oper. Res. 2023;304(3):865–86.

[16] Wang Y, Ma X, Li Z, Liu Y, Xu M, &Wang Y. Profit distribution in collaborative multiple centers vehicle routing problem. J. Clean. Prod. 2017;144:203–19.

[17] Dellaert N, Woensel TV, Crainic TG, Saridarq FD. A multi-commodity two-Echelon capacitated vehicle routing problem with time windows: Model formulations and solution approach. Comput. Oper. Res. 2021;127:105154.

[18] Jia S, Deng L, Zhao Q, Chen Y. An adaptive large neighborhood search heuristic for multi-commodity two-echelon vehicle routing problem with satellite synchronization. J. Ind. Manage. Optim. 2022;19(2):1187–210.

[19] Gu W, Archetti C, Cattaruzza D, Ogier M, Semet F, Speranza MG. A sequential approach for a multi-commodity two-echelon distribution problem. Comput. Ind. Eng. 2021:107793.

[20] Li H, Zhang L, Lv T, Chang X. The two-echelon time-constrained vehicle routing problem in linehaul-delivery systems. Transport. Res. Part B Methodolog. 2016;94:169–88.

[21] Dellaert N, Dashty Saridarq F, Van Woensel T, Crainic TG. Branch-and-price–based algorithms for the two-echelon vehicle routing problem with time windows. Transport. Sci. 2019;53(2):463–79.

[22] Anderluh A, Nolz PC, Hemmelmayr VC, Crainic TG. Multi-objective optimization of a two-echelon vehicle routing problem with vehicle synchronization and "grey zone" customers arising in urban logistics. Eur. J. Operat. Res. 2021;289(3):940–58.

[23] Belgin O, Karaoglan I, Altiparmak F. Two-echelon vehicle routing problem with simultaneous pickup and delivery: Mathematical model and heuristic approach. Comput. Ind. Eng. 2018;115:1–16.

[24] Li J, Xu M, Sun P. Two-echelon capacitated vehicle routing problem with grouping constraints and simultaneous pickup and delivery. Transportation Research Part B-Methodological 2022;162:261–91.

[25] Paul A, Kumar RS, Rout C, Goswami A. A bi-objective two-echelon pollution routing problem with simultaneous pickup and delivery under multiple time windows constraint. Opsearch 2021;58(4):1–32.

[26] Zhou H, Qin H, Zhang Z, Li J. Two-echelon vehicle routing problem with time windows and simultaneous pickup and delivery. Soft. Comput. 2022;26(7):3345–60.

[27] Wang Y, Peng S, Xu C, Assogba K, Wang H, Xu M, Wang Y. Two-echelon logistics delivery and pickup network optimization based on integrated cooperation and transportation fleet sharing. Expert. Syst. Appl. 2018;113:44–65.

[28] Li H, Wang H, Chen J, Bai M. Two-echelon vehicle routing problem with satellite bi-synchronization. Eur. J. Oper. Res. 2021;288(3):775–93.

[29] Wang Y, Li Q, Guan X, Xu M, Liu Y, Wang H. Two-echelon collaborative multi-depot multi-period vehicle routing problem. Expert. Syst. Appl. 2021;167:114201.

[30] Wang Y, Zhang S, Assogba K, Fan J, Xu M, Wang Y. Economic and environmental evaluations in the two-echelon collaborative multiple centers vehicle routing optimization. J. Clean. Prod. 2018;197:443–61.

[31] Fan H, Zhang Y, Tian P, Lv Y, Fan H. Time-dependent multi-depot green vehicle routing problem with time windows considering temporal-spatial distance. Comput. Oper. Res. 2021;129:105211.

[32] Sherif S, Asokan P, Sasikumar P, Mathiyazhagan K, Jerald J. Integrated optimization of transportation, inventory and vehicle routing with simultaneous pickup and delivery in two-echelon green supply chain network. J. Clean. Prod. 2021;287:125434.

[33] Zhen L, Ma C, Wang K, Xiao L, Zhang W. Multi-depot multi-trip vehicle routing problem with time windows and release dates. Transport. Res. Part E Logist. Transport. Rev. 2020;135:101866.

[34] Wang Y, Li Q, Guan X, Fan J, Xu M, Wang H. Collaborative multi-depot pickup and delivery vehicle routing problem with split loads and time windows. Knowl. Based. Syst. 2021;231:107412.

[35] Soriano A, Gansterer M, Hartl R. The multi-depot vehicle routing problem with profit fairness. Int. J. Prod. Econ. 2023;255:108669.

[36] Xue G. A two-stage heuristic solution for multi-depot collaborative pickup and delivery network with transfers to reduce carbon emissions. J. Clean. Prod. 2022;373:133539.

[37] Zhang Q, Wang Z, Huang M, Yu Y, Fang S. Heterogeneous multi-depot collaborative vehicle routing problem. Transportation Research Part B-Methodological 2022;160:1–20.

[38] Berbeglia G, Cordeau JF, Laporte G. Dynamic pickup and delivery problems. Eur. J. Oper. Res. 2010;202(1):8–15.

[39] Ropke S, Pisinger D. An adaptive large neighborhood search heuristic for the pickup and delivery problem with time windows. Transport. Sci. 2006;40(4):455–72.

[40] Alinaghian M, Shokouhi N. Multi-depot multi-compartment vehicle routing problem solved by a hybrid adaptive large neighborhood search. Omega (Westport) 2018;76:85–99.

[41] Azi N, Gendreau M, Potvin JY. An adaptive large neighborhood search for a vehicle routing problem with multiple routes. Comput. Oper. Res. 2014;41:167–73.

[42] Franceschetti A, Demir E, Honhon D, Van Woensel T, Laporte G, Stobbe M. A metaheuristic for the time-dependent pollution-routing problem. Eur. J. Oper. Res. 2017;259(3):972–91.

[43] Ribeiro GM, Laporte G. An adaptive large neighborhood search heuristic for the cumulative capacitated vehicle routing problem. Comput. Oper. Res. 2012;39(3):728–35.

[44] Sacramento D, Pisinger D, Ropke S. An adaptive large neighborhood search metaheuristic for the vehicle routing problem with drones. Transport. Res. Part C Emerg. Technol. 2019;102:289–315.

[45] Potvin JY, Kervahut T, Garcia BL, Rousseau JM. A tabu search heuristic for the vehicle routing problem with time windows. Quebec, Canada: Universite de Montreal; 1992. Technical Report CRT-855.

[46] Demir E, Bektas¸ T, &Laporte G. An adaptive large neighborhood search heuristic for the Pollution-Routing Problem. Eur. J. Oper. Res. 2012;223(2):346–59.

[47] Nagata Y, Bräysy O, Dullaert W. A penalty-based edge assembly memetic algorithm for the vehicle routing problem with time windows. Comput. Oper. Res. 2010;37(4):724–37.

[48] Cordeau J, Laporte G, Mercier A. Improved tabu search algorithm for the handling of route duration constraints in vehicle routing problems with time windows. J. Oper. Res. Soc. 2004;55(5):542–6.

[49] Sadati M, Catay B, Aksen D. An efficient variable neighborhood search with tabu shaking for a class of multi-depot vehicle routing problems. Comput. Oper. Res. 2021;133:105269.

[50] Breunig U, Schmid V, Hartl RF, Vidal T. A large neighbourhood based heuristic for two-echelon routing problems. Comput. Oper. Res. 2016;76:208–25.