Logist. Res. (2011) 3:101–120
DOI 10.1007/s12159-011-0046-9

ORIGINAL PAPER

A multi-agent and auction-based framework and approach for carrier collaboration

Bo Dai · Haoxun Chen

Received: 3 September 2010 / Accepted: 17 January 2011 / Published online: 11 February 2011

Abstract Carrier collaboration in transportation means multiple carriers form an alliance to optimize their transportation operations through sharing transportation requests and vehicle capacities. In this paper, we propose a multi-agent and auction-based framework and approach for carrier collaboration in less than truckload transportation. In this framework, the carriers outsource/acquire requests through multiple auctions, one for outsourcing each request; a carrier acts as an auctioneer when it wants to outsource a request to other carriers, whereas the carrier acts as a bidder when it wants to acquire a request from other carriers; for each carrier, which requests it should outsource and acquire are determined by solving its outsourcing requests selection problem and requests bidding problem, respectively. These two decision problems are formulated as mixed integer programming problems. The auction of each request is multiround; in each round, the auctioneer determines the outsourcing price of the request and each bidder determines whether it acquires the request at the given price; the auctioneer lowers the outsourcing price if multiple carriers bid for the request or raises the price if no carrier bids for it. The auction process continues until only one carrier bids for the request or a given number of rounds are achieved. In the second case, if no agent bids for the request, then it is returned to the outsourcing agent; if multiple bidding agents compete for the request, a conflict resolution procedure is used to determine which carrier wins it. The approach is decentralized, asynchronous, and dynamic, where multiple auctions may occur simultaneously and interact with each other. The performance of the approach is evaluated by randomly generated instances and compared with an individual planning approach and a centralized planning approach.

Keywords Collaborative transportation planning · Carrier collaboration · Multi-agent systems · Auction · Outsourcing · Pricing

1 Introduction

In collaborative logistics, multiple carriers or shippers may form an alliance to optimize their transportation operations by sharing vehicle capacities and delivery requests. The objective of the collaboration is to reduce empty backhauls, to raise vehicle utilizations, and thus to increase the profit of each carrier involved. In practice, two types of transportation services are often provided: truckload (TL) transportation and less than truckload (LTL) transportation. The two types have respective application domains and advantages. TL is often used in the transportation of a single product of large quantity from a shipper to a customer, whereas LTL is usually used to transport one or multiple products in small volumes from multiple shippers to multiple customers such as in parcel delivery. The main advantage of LTL is that a shipment may be performed with a fraction of the cost of hiring an entire vehicle or trailer for an exclusive shipment. A number of accessorial services are available for LTL, which are not typically offered by TL [25]. In case that customers demand frequent deliveries in small quantities in order to reduce their

B. Dai · H. Chen (✉)
Laboratoire d'optimisation des Systèmes Industriels (LOSI),
Institut Charles Delaunay (ICD) and UMR CNRS STMR 6279,
Université de Technologie de Troyes, 12 rue Marie Curie,
BP 2060, 10010 Troyes Cedex, France
e-mail: haoxun.chen@utt.fr

B. Dai
e-mail: bo.dai@utt.fr

inventory costs, LTL shipment must be used by carriers to increase their vehicle utilization rates. That is why LTL became more popular in recent years. Two types of collaboration are used. One is shipper collaboration, which focuses on the collaboration among multiple shippers whose transportation requests are served by a single carrier. Through collaboration, the shippers may be able to identify and submit routes with less asset reposition to the carrier for obtaining more favorable rates. The other is carrier collaboration that considers the reassignment of transportation requests among carriers in order to reduce their total transportation costs for serving the requests. The cost savings as a result of the collaboration will be shared among the carriers. Two key issues for carrier collaboration are optimal reassignment of transportation requests among carriers to maximize their total profit and fair allocation of the profit among the carriers. In this paper, we study carrier collaboration in less than truckload transportation with pickup and delivery requests and propose a multi-agent and auction-based framework and approach for the problem. The literature relevant to our work is first reviewed in the following.

1.1 Collaboration in transportation

Previous studies on collaborative transportation can be classified into two categories according to whether the considered transportation service is TL or LTL. For TL transportation, most papers adopt a centralized planning approach in which profit/cost allocation among collaborative partners is a key issue. Ergun et al. [9, 10] studied shipper collaboration and formulated its centralized planning problem as a lane covering problem; Houghtalen et al. [11] studied carrier collaboration and proposed a mechanism for allocating both resources and profits among carriers by appropriately setting prices for the resources. Kwon et al. [15] and Lee et al. [18] proposed a combinatorial auction mechanism for transportation procurement of a shipper from carriers. Agarwal and Ergun [1] considered carrier collaboration in a multicommodity service network and designed a mechanism for allocating benefits among carriers based on a decentralized multicommodity flow game, capacity exchange costs, and inverse optimization.

For LTL transportation, Dai and Chen [6, 7] proposed a centralized framework for the problem without time windows for pickup and delivery operations of requests, and the global optimal transportation planning problem for the carrier alliance was formulated as a mixed integer programming model. To solve the model, a Lagrangian relaxation approach and a revised benders decomposition approach were developed. The performances of the solution approaches were evaluated by randomly generated instances. Krajewska and Kopfer [13] presented a decentralized combinatorial auction model for the collaboration among independent freight forwarding entities. Their collaboration model includes three phases: preprocessing, profit optimization, and profit sharing. In the preprocessing phase, each carrier specifies its lowest cost for fulfillment of each request outsourced to its collaboration partners. In the profit optimization phase, the outsourcing requests are reallocated to the collaborating partners. In the profit sharing phase, the profit gained from the fulfillment of each outsourcing request is allocated among the coalition members according to a collaboration advantage index. Krajewska et al. [14] considered profit sharing in a horizontal cooperation (collaboration) among freight carriers, with customer requests for each carrier as those of a pickup and delivery problem with time windows. The profit margins resulting from the cooperation were analyzed, and the possibilities of sharing these profit margins fairly among the carriers were discussed. The Shapley value in cooperative game theory was used to determine a fair profit allocation. Berger and Bierwirth [3] utilized decentralized control and auction-based exchange mechanisms for the request reassignment problem in collaborative carrier networks. A five-step procedure was proposed for the request reassignment: forming a request candidate set, composition of bundles from the candidate set, determination of marginal profits, assignment of bundles to carriers, and profit sharing. They compared their decentralized method with a centralized method by numerical experiments. Their approach belongs to the category of single-round auctions. Moreover, they ignored the vehicle capacity constraint by assuming that all shipments take only a very small fraction of the capacity of a vehicle.

1.2 Multi-agent systems in transportation

A system with multiple autonomous agents interacting with each other is called a multi-agent system (MAS). It is a new paradigm for modeling and studying distributed systems. In an MAS, all agents are autonomous computational components that are able to control their own behaviors in the furtherance of their own goals [31]. They can cooperate, coordinate, and negotiate with each other as human beings. Davidsson et al. [8] provided a survey of existing research on agent-based approaches for transportation and traffic management with the focus on freight transportation. They concluded that agent-based approaches seem very suitable for this domain but need to be verified by more deployed systems. Lang et al. [16] gave a literature review on the application of MAS in logistics with the emphasis on the role of MAS coordination architectures. They concluded that planning problems in transportation have characteristics complying with the specific capabilities of agent systems and their capability of dealing with inter-organizational and event-driven planning settings meets requirements in supply chain planning and execution.

For the application of MAS in collaborative logistics, Mes and Van der Heijden [20] and Mes [21] used a multi-agent system to model a transportation marketplace, where shippers offer time-sensitive full truckload pickup and delivery jobs through sequential auctions and carriers compete with each other to service these jobs. To cope with the interdependencies among jobs, they used reserve prices and decommitment penalties and proposed two strategies for shippers to maximize their revenue in sequential auctions, namely delaying and breaking commitments. The idea of delaying commitments is that a shipper will not agree with the best bid whenever it is above a certain reserve price. The idea of breaking commitments is that the shipper allows the carriers to break commitments against certain penalties. In this way, the pricing and scheduling decisions of carriers can take into account both the direct costs of jobs and their impact on future opportunities [21].

1.3 Auctions in transportation

An auction is a protocol that allows agents to indicate their interests in one or more resources and that uses these indications of interest to determine both an allocation of resources and an allocation of payments among the agents. In the literature, different types of auctions were proposed, which include single-good auctions, multiunit auctions, combinatorial auctions, and exchanges [24]. As an important category of auction, exchanges signify that agents are able to act both as buyers and sellers. One important type of auction in exchanges is called two-sided auction or double auction. A double auction is associated with a single-dimensional double market, where multiple potential buyers and sellers of multiple units of the same good exist. The double auctions are further classified into two types, continuous double auction (CDA) and periodic double auction (call market). In combinatorial auctions, each bidder may bid not only for a (single) combination of items but also for any subset of all combinations of items. The combinatorial auction problem (CAP) that determines the winners of the bids is usually formulated as a NP-hard set packing problem (SPP). Two kinds of combinatorial auctions are studied in this literature: single-round auction and multirounds auction (iterative auction). The multirounds auction has several advantages over the single-round auction [15, 27]. Iterative auctions have two types of modes: quantity-setting auctions and price-setting auctions. In each round of a quantity-setting auction, bidders send their valuations over the items they want to buy. The auctioneer makes a provisional allocation that depends on the submitted prices, and bidders adjust the prices. In each round of a price-setting auction, each bidder submits a bundle of items it wants to get at the given prices. The auctioneer adapts the prices to balance the supply and demand [26]. Price-setting auctions and quantity-setting auctions are dual to each other. Price-setting auctions correspond to the primal–dual algorithms of CAP.

Auctions were used in transportation planning early in Wellman [29]. The author proposed a general "market-oriented programming" environment (approach) to the construction and analysis of distributed transportation planning systems, based on general equilibrium theory and competitive mechanisms. The environment provides basic constructs for defining computational market structures and a procedure for deriving their corresponding competitive equilibria. Sandholm [22] presented a formalization of the bidding and awarding decision process for the contract net-based task allocation. This formalization is based on marginal cost calculations based on local agent criteria. In the proposed model, agents having very different local criteria (based on their self-interests) can interact with distribute tasks so that the network of the agents as a whole operates more effectively. The contract net protocol was verified by transportation cooperation net, where dispatch centers of different companies cooperated automatically in vehicle routing. Bürckert et al. [5] described a multi-agent dispatching system developed in cooperation with a forwarding company. The system can deal with dynamical arrival of transportation requests. The novelty of the system is that it uses holonic agents, which are composed of subagents that interact in a cooperative way.

Combinatorial auction has been applied to truckload transportation service procurement. Kwon et al. [15] proposed an integrated multiround combinatorial auction mechanism, where shippers allow bids on packages of lanes (requests). In each round, each carrier must solve a bid generation problem (BDP, [18] to discover the most profitable bundle of lanes that maximizes its profit, and the auctioneer must solve a winner determination problem (WDP) to assign bundles of lanes to carriers. The price information derived from the solution of WDP in each round is used by bidders to identify the most profitable requests in next round. The mechanism integrates the shipper and bidder optimizations. The numerical results indicate that better allocations are obtained for both shippers and carriers, i.e., shippers reduce costs of procurement of services and carriers are able to identify alternative valuable packages of lanes. However, BDP is a non-linear integer programming problem and WDP is a 0–1 integer programming problem, both of them are very hard to solve for large instances.

For combinatorial auctions applied to carrier collaboration in LTL transportation, Schwind et al. [23] designed a combinatorial exchange mechanism ComEx for the inter-division exchange of delivery orders in a logistics company organized in a profit center structure. The exchange mechanism is subdivided into four phases: initialization phase, outsourcing phase, insourcing phase, and final evaluation phase. The outsourcing phase and the insourcing phase determine, respectively, the delivery orders to be outsourced to other profit centers and the delivery orders to be acquired from other profit centers. In the final evaluation phase, a combinatorial auction is performed by finding the cost-minimal allocation of order clusters to the profit centers. To further reduce the delivery costs of the entire logistics company, an iterative auction mechanism is also proposed. Their numerical experiments demonstrate that using ComEx, the total transportation cost of the logistics company can be reduced by up to 14%.

In addition, Wellman et al. [30] studied auction protocols for a decentralized scheduling problem. They investigated the existence of equilibrium prices for some general classes of scheduling problems, the quality of equilibrium solutions, and the behavior of an ascending auction mechanism and bidding protocol. Each task in the scheduling problem corresponds to an ascending auction. All auctions involved are performed asynchronously. Walsh and Wellman [28] studied the supply chain formation problem. A market price system about the resources produced along the chain is introduced to decentralize the formation process. In a competitive equilibrium of this system, agents select local optimal allocations based on prices, leading to a globally optimal allocation. A market protocol based on distributed, progressive auctions, and myopic, non-strategic agent bidding policies is used to determine prices. The price of each resource is determined by an ascending auction similar to what used in Wellman et al. [30]. The protocol produced better solutions than the greedy protocols common under resource contention.

In this study, we choose MAS and auction as two major components of our approach because: (1) Each carrier in carrier collaboration is an autonomous unit with its own private information and decision-making authority; (2) As pointed out by Lang et al. [16], transportation firms engage in a high level of negotiation and cooperation in performing their daily transport tasks, and MAS can model such cooperative capabilities; and (3) It is in the nature of auctions to address the reassignment of transportation requests and the allocation of the profit gained by carrier collaboration simultaneously. Instead of designing a post-collaboration profit allocation mechanism, we propose a multi-agent and auction-based collaboration framework whose profit allocation among carriers is determined by the auction process of each request. So far, most studies using combinatorial auction for carrier collaboration adopted a single-round auction, which requires preselection of preferable bundles of requests from an exponential number of bundles by each carrier and the resolution of a NP-hard winner determination problem by the auctioneer. The price-setting-based auction can avoid the two difficulties, which is adopted in our framework.

In our framework, each carrier is regarded as an autonomous agent, and the collaboration among carriers is realized by auction. As a self-governed agent, it determines its requests to be outsourced to other carriers and set the price for each outsourcing request in each iteration of the auction. The auction of each outsourcing request organized by an agent belongs to the category of exchange or double auction, in which each agent acts both as an auctioneer and as a bidder. For an outsourcing request, its auction process starts when the offering agent acts as an auctioneer who announces the outsourcing of the request with its initial outsourcing price. Each of the other agents then acts as a bidder and expresses its attitude to this request (bid or not for the request). If the auctioneer does not receive any bid after a given time period corresponding to a round of auction, that is, no bidder is willing to acquire the request, the auctioneer will increase the outsourcing price. Otherwise, if more than one agents bid for the request, the auctioneer may lower the price. The price update is similar to what in some well-known single-good auctions. There are two stopping conditions for terminating the auction process: only one agent bids for the request or a given number of rounds are achieved. In the former case, the request will be allocated to the unique bidder. The latter case indicates that no agent is interested in the request or multiple agents compete for the same request. In the second situation, a conflict resolution procedure will be used to assign the request to one of the bidding agents. The framework is decentralized, asynchronous, and dynamic, where multiple auctions may occur simultaneously and interact with each other. In the framework, two decision problems for each carrier are the outsourcing requests selection problem that determines non-profitable requests to be outsourced and the requests bidding problem that determines requests to be acquired from other carriers. Mixed integer programming models are provided for the two problems.

The previous studies relevant to ours are Mes and Van der Heijden [20], Mes [21], Wellman et al. [30], and Walsh and Wellman [28]. Compared with sequential auctions for truckload pickup and delivery jobs proposed in Mes and Van der Heijden [20] and Mes [21], our proposed auction is not sequential but asynchronous. In our auction, each agent (carrier) may engage in multiple auctions at the same time. Moreover, in their auctions, the interdependencies among jobs (objects) are coped by using reserve prices and de-commitment penalties, whereas our auction takes into account such interdependencies by considering all open auctions in outsourcing requests selection problem and requests bidding problem of each carrier (bidder as well as auctioneer). Finally, our auction is applied to carrier collaboration in less than truck load transportation rather than shippers–carriers collaboration in full truck load transportation dealt with their auctions. Both our proposed auction and the auctions proposed in Wellman et al. [30] and Walsh and Wellman [28] run asynchronously without direct coordination. However, our auction distinguishes from theirs with the following three aspects: (1) Our auction permits the ascending and descending of the price of each object (request), whereas theirs only allow the ascending of the price of each object; (2) In our auction, each bidder only determines whether to acquire a request (object) at its price currently announced by the corresponding auctioneer, whereas each bidder in their auctions has to determine the bidding price for each bid in each round; (3) Our auction is applied to carrier collaboration in collaborative logistics, whereas theirs are applied to decentralized scheduling and supply chain formation, respectively. In addition, our proposed approach is quite different from that proposed by Berger and Bierwirth [3]. The former is an asynchronous multiround (iteration) auction approach in which multiple auctions may occur simultaneously and each carrier may act both as an auctioneer and as a bidder, whereas the latter is a (synchronous) single-round combinatorial auction approach in which only one auction is performed (with a single auctioneer). The advantages of our price-based iterative auction approach beyond their approach are that our approach neither needs to select a set of preferable bundles of requests from an exponential number of bundles of requests for each bidder (carrier) nor needs to solve a NP-hard combinatorial auction problem (CAP) to determine the assignment of bundles to carriers. Besides, our problem and their problem have different settings; for instance, the capacity of each vehicle is ignored in their problem by assuming that the shipments take only a very small fraction of its capacity, whereas we do consider the vehicle capacity.

The novelty and contribution of our work are mainly in four aspects: (1) In our multi-agent and auction-based framework, each carrier keeps its decision autonomy, no central coordinator (auctioneer) is required, and all the auctions operate asynchronously and are driven by individual agents; (2) No private information is shared among agents, and the framework is entirely decentralized; (3) The decentralized decision making in the framework can reduce computational complexity and is thus well suitable for dynamic collaboration environments where transportation requests of each carrier arrive dynamically, the executions of the requests finish dynamically, and the availability of resources of each carrier changes dynamically. (4) Through employing price-setting-based auctions in our framework, each carrier neither needs to select its preferable bundle from an exponential number of bundles nor needs to determine a price for each bundle. The performance of the proposed framework and approach is evaluated by simulation and is compared with the case of no collaboration among agents and with a centralized framework and approach. The simulation results show that by reducing transportation costs, our proposed decentralized collaboration mechanism can increase the total profit of all carriers as well as each agent's individual profit.

The rest of this paper is organized as follows. Section 2 describes the carrier collaboration problem studied and Sect. 3 outlines our multi-agent and auction-based framework. Section 4 formulates the outsourcing requests selection problem, which includes a method for setting an initial outsourcing price for each outsourcing request. Section 5 introduces the requests bidding problem. Section 6 presents the asynchronous multi-agent multi-auction process. Section 7 evaluates our proposed framework and approach by simulation and compares it with a centralized framework and approach, where random instances are generated for the evaluation and comparison. Section 8 concludes this paper by some remarks on future research.

2 Problem description

In this paper, we study a carrier collaboration problem in less than truckload transportation with pickup and delivery requests (CCPLTL). One application of such problem is parcel delivery. To be general, we do not specify particular product types to transport in this problem.

In the problem, multiple carriers with a depot and a limited number of capacitated vehicles operate in a common transportation network. Initially, each carrier has collected certain transportation requests from its customers (shippers) with a price for serving each request paid by a customer to the carrier (a request collected by a carrier from its shipper is also called a request of the carrier hereafter). Each request is specified by a pickup location and time window, a quantity, and a delivery location and time window. To increase vehicle utilization rates and reduce empty backhauls, all carriers form a collaborative alliance that shares vehicle capacities among them. For each carrier, its collected requests are not always profitable, it may outsource non-profitable requests to other carriers in the alliance, and all outsourcing requests are available to all carriers. If a request is not served by its offering carrier, then the offering carrier gains the difference between the price paid by its customer for serving the request and the outsourcing price of the request determined by the carrier.

It is assumed that each carrier wants to assure a "minimum profit margin" [12]. The minimum profit margin of a carrier represents its profitability expectation. The higher the expectation, the larger the minimum profit margin. Different carriers may have different minimum profit margins. In our problem, the profit of the carrier is defined as its transportation revenue minus its transportation cost. The minimum profit margin plays an important role in determining whether the carrier will serve a request demanded by a shipper, whether it will outsource a request to other carriers, and what should be the outsourcing price of the request. One important decision problem for a carrier is to set its minimum profit margin. If the carrier accepts all requests that are profitable, its minimum profit margin is set to zero. Otherwise, if the carrier accepts only high profitable requests, its minimum profit margin should be set to a large percentage. According to an investigation of Holguín-Veras et al. [12], the profit of a carrier is determined by its direct costs and the profit margin it selects. A minimum value of the profit margin of 5% was assumed as the opportunity cost of the capital. For this reason, most carriers chose 5% as their minimum profit margin in order to win a bid, which is what economic theory would predict in a competitive market, i.e., rates set at marginal costs. Of course, in reality, each carrier can freely set its minimum profit margin according to its own profitability expectation. In our study, we assume that the minimum profit margin of each carrier is given.

In the studied carrier collaboration problem, the objective of each carrier is to maximize its own profit, while the collaboration among carriers is achieved by exchanging transportation requests among them. The profit is obtained by subtracting the total transportation cost from the total revenue for serving the requests. A solution of the problem is given by a set of requests to be served by each carrier, a set of optimal vehicle tours for the carrier to serve the set of requests, and the outsourcing price of each request outsourced by each carrier to other carriers. Note that the set of requests to be served by a carrier includes the requests it collects from its shippers and serves by itself and the requests it acquires from other carriers. Multiple requests may be served by the same vehicle tour. For each carrier, its feasible transportation plan is defined by a set of vehicle tours; each vehicle tour leaves from and returns to its depot; the load of each vehicle does not exceed its capacity; for each request served, its pickup operation is performed before its delivery operation; and each pickup or delivery operation is performed within the time window. In the problem, the transportation network considered is represented by a directed graph $D = (N, A)$ with node set N and arc set A. Each arc is associated with a traveling time which is the length of the arc, which may be different for different carriers. So different carriers may have different transportation costs associated with each arc. Other parameters of the problem include the following: the number of carriers; the coordinates of each node and the length of each arc in the transportation network; the number of vehicles of each carrier; the capacity of every vehicle of each carrier; the depot node of each carrier; the transportation cost between any two nodes for every vehicle of each carrier; the set of requests collected by each carrier from its shippers; the price paid by shippers to carriers to serve each request, pickup node, delivery node, quantity, pickup time window, and delivery time window of each request; the minimum profit margin of each carrier. The decision variables of the problem for each carrier include: the set of requests served by itself, the set of requests to be acquired from other carriers and served by itself, a set of vehicle tours to serve the two sets of requests, the outsourcing price and the winner of each request outsourced by the carrier.

To better understand CCPLTL described above, we design an illustrative example that has a transportation network with 21 nodes, three carriers a, b, c with nodes 5, 17, 11 as their depots, respectively. Each carrier has 10 vehicles with the same capacity $C = 10$. The transportation network is given in Fig. 1. The total number of transportation requests of the carriers is nine; the requests are represented by $r1$ to $r9$, respectively. For each request, its quantity is indicated by a number associated with the arc representing the request. For example, request $r1$ is to pick up five quantity of freight at node 21 and deliver this quantity to node 13. Carrier a has requests $r1$, $r2$, $r3$, carrier b has requests $r4$, $r5$, $r6$, and carrier c has requests $r7$, $r8$, $r9$. The coordinates of each node in the transportation network is given in Table 1, and the time windows and the price paid by a shipper for serving each request are given in Table 2. The time window of each depot node is [0, 240].

3 Multi-agent and auction-based framework

Motivated by multi-agent system models applied to scheduling problems [2] and the combinatorial auction mechanism proposed for truckload transportation service procurement [15, 18], we propose a multi-agent and auction-based framework for CCPLTL. In the framework, each carrier is an autonomous agent with decision authority, and the interactions between carriers are realized through multiple auction processes of outsourcing requests. Since each carrier both outsources (sells) and acquires (buys) requests, it acts both as an auctioneer and as a bidder in auction terms.

For each request that is outsourced by an agent (carrier), its auction process starts when the agent announces the outsourcing of the request together with its initial outsourcing price, where the agent is an auctioneer. After the auction is started, if the auctioneer does not receive any reply for the request from a bidder (carrier) for a given time

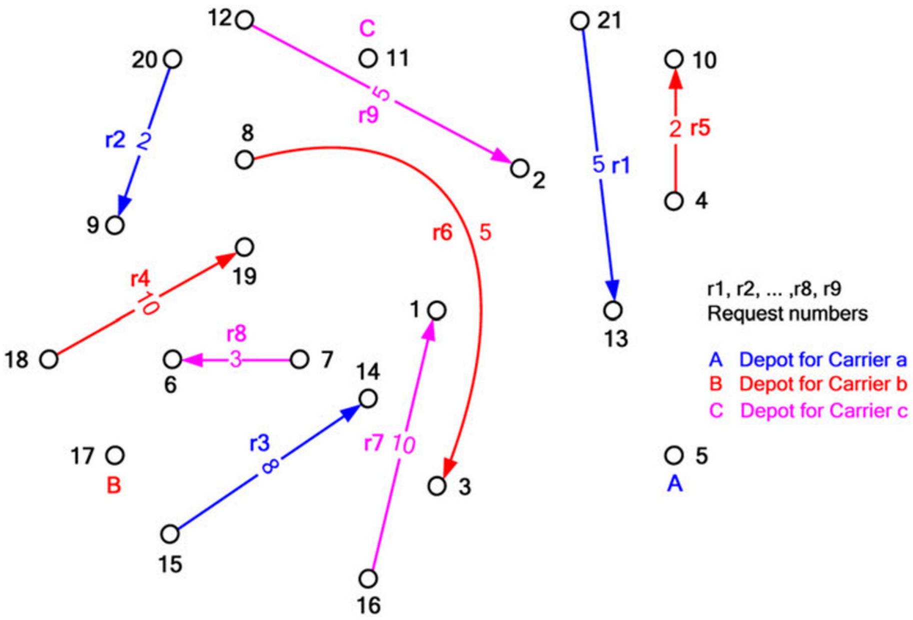

Fig. 1 Transportation network of the illustrative example

Table 1 Coordinates of the nodes in the transportation network

Node	1	2	3	4	5	6	7	8	9	10	11
X	35	41	35	55	55	15	25	20	10	55	30
Y	35	49	17	45	20	30	30	50	43	60	60
Node	12	13	14	15	16	17	18	19	20	21	
X	20	50	30	15	30	10	5	20	15	45	
Y	65	35	25	10	5	20	30	40	60	65	

Table 2 Time windows, quantity, and service price of the requests

Request	1	2	3	4	5	6	7	8	9
Pickup node	21	20	15	18	4	8	16	7	12
Delivery node	13	9	14	19	10	3	1	6	2
Pickup time window	[97,122]	[109,147]	[112,144]	[123,161]	[91,144]	[127,172]	[129,173]	[159,188]	[68,127]
Delivery time window	[139,193]	[115,130]	[120,141]	[166,221]	[106,130]	[168,195]	[127,182]	[162,193]	[99,140]
Quantity	5	2	8	10	2	5	10	3	5
Service price	129	70	197	127	63	115	259	52	62

period determined by the auctioneer, the bidder is considered not willing to acquire the request. After the given time is elapsed for the request, the auctioneer can lower the outsourcing price if more than one agents bid for the request or raise the price if no agent bids for the request. The price update is similar to what in some well-known single-good auctions, which will be introduced in Sect. 6. The process continues until one of two stopping conditions is satisfied: only one agent bids for the request or a given number of iterations are achieved. If the process is terminated by the second condition, either no agent wants to acquire the request or multiple agents compete for the same request at the end of the auction. In the latter case, a conflict resolution procedure, which will be introduced in Sect. 6, is used to determine who wins the request. The interactions among multiple agents are illustrated in Fig. 2.

The proposed framework (multi-agent system) is decentralized, asynchronous, iterative, and dynamic. Here, “decentralized” means that each agent makes its own decision; “asynchronous” means all auction processes happened are asynchronous, there is no order among outsourcing requests selection, bid generation, and price adjustment of different carriers, and the only relationship among the carriers is the exchange of information on

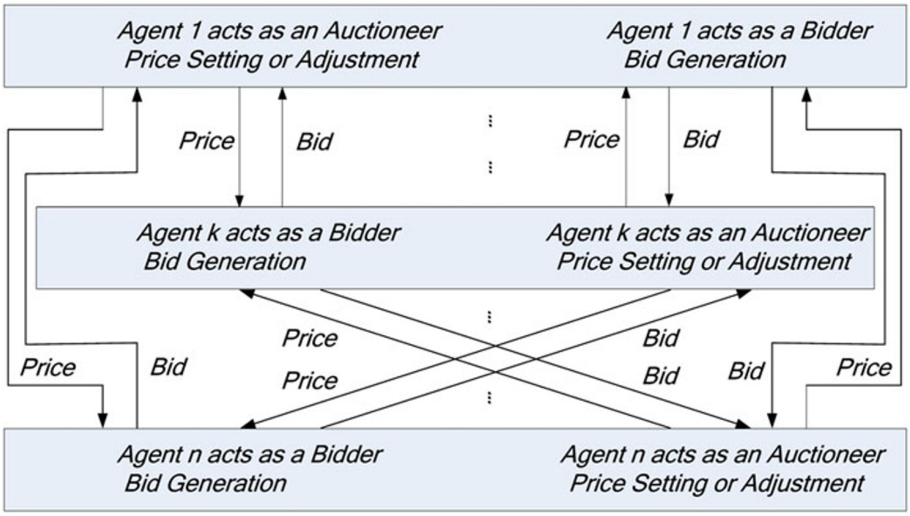
Agent 1 acts as an Auctioneer
Price Setting or Adjustment
Agent 1 acts as a Bidder
Bid Generation
Price
Bid
Price
Bid
Agent k acts as a Bidder
Bid Generation
Agent k acts as an Auctioneer
Price Setting or Adjustment
Price
Bid
Price
Price
Bid
Bid
Bid
Price
Agent n acts as a Bidder
Bid Generation
Agent n acts as an Auctioneer
Price Setting or Adjustment

Fig. 2 The interactions among multiple agents

requests outsourcing and acquisition among them; "iterative" means that the auction process of each outsourcing request is a multirounds iterative process; and "dynamic" means that the multi-agent system is a dynamic system, new requests arrive dynamically, the execution of requests finishes dynamically, and the availability of resources (vehicles) of each carrier may change dynamically, etc.

The dynamic behavior of the multi-agent system can be described by the evolution of its state over time. The state of the system consists of the states of all carriers. The state of a carrier is represented by the states of all requests concerning the carrier and the states of the resources (vehicles) of the carrier. For a self-executed request, it has three possible states: *before execution*, *in execution*, and *executed*. On the other hand, for an outsourcing request, it has five possible states: *before auction*, *in auction*, *auctioned* (auction is completed), *in execution*, and *executed*. For each outsourcing request, which agent (carrier) will execute, it is determined as soon as its auction process is finished. Such request is executable only when its auction process is completed. An outsourcing request pool is introduced in our framework to gather all outsourcing requests that are announced by all agents and are currently in auction. All requests in the pool are available to each agent. In the framework, except for the price adjustment (update) of outsourcing requests by agents, two decision problems for each carrier are involved: one is to determine non-profitable requests to be outsourced and the other is to determine requests to be acquired from other carriers. These two decision problems are referred to as outsourcing requests selection problem (ORSP) and requests bidding problem (RBP), respectively, hereafter.

When an agent has chosen to outsource a request, the agent must determine at the same time an initial outsourcing price for the request. The price will be adjusted (updated) by the agent during the auction process of the request. When an agent selects a set of requests to be outsourced, it must consider its uncompleted requests that are collected from its shippers (self-executed requests with state *before execution* or *in execution*), already acquired other agents' outsourcing requests that have not been completed (outsourcing requests with state *before execution* or *in execution*), and probably, the requests in the current outsourcing request pool if the agent determines its requests outsourcing and acquisition jointly. A request can be outsourced only if it is at the state of *before execution*.

Each agent also determines the requests acquired from other agents. A request can be acquired by an agent only if it is an outsourcing request at the state of *in auction*. When an agent acts as an auctioneer and announces the auction of an outsourcing request, each of the other agents, which is a bidder, can reply to this outsourcing announcement by choosing whether it acquires (bids for) the request or not. When an agent chooses to acquire a set of outsourcing requests, it must consider its uncompleted requests that are collected from its shippers (self-executed requests with state *before execution* or *in execution*), already acquired other agents' outsourcing requests that have not been completed (outsourcing requests with state *before execution* or *in execution*) and the requests in the outsourcing request pool. Once an agent is awarded an outsourcing request, this allocation is irreversible in later rounds.

Compared with other auction-based frameworks previously proposed for the carrier collaboration, our proposed framework has several new features and advantages: (1) Each carrier is totally autonomous, which makes its own decisions, there is no central auctioneer (coordinator) that makes decisions on behalf of the carriers; each carrier can play both the role of an auctioneer and the role of a bidder; (2) The price paid by a shipper to a carrier for serving a request is private information reserved only by the carrier and not disclosed to other carriers; (3) In our price-setting framework, each carrier does not need to select its preferable bundle from an exponential number of bundles (2^n for n requests). What each carrier should do is to determine a set of requests to be outsourced by solving an outsourcing requests selection problem and a set of requests to bid by solving a requests bidding problem based on the current price of each request. They do not need to determine a price for executing a bundle of requests; (4) In our framework, carriers act as both buyer and seller and the price of each resource (request) is determined (adjusted) by its demand and supply as in a market economy, and the prices may be increased and decreased according to the bidders' attitudes.

4 Outsourcing requests selection problem

To initiate an auction-based collaboration process among carriers, each carrier must select a set of requests to be outsourced from its own requests. Our requests outsourcing model for each carrier is based on the idea of minimum profit margin as explained in Sect. 2. For a carrier considered, suppose that its minimum profit margin is α ($0 < \alpha < 1$). For a request l of the carrier, let p_l denotes the price paid by a shipper to the carrier for serving request l, that is, p_l is the revenue obtained by the carrier if it serves the request. Since the carrier wants to achieve a minimum profit margin α, its least profit to gain by serving the request is αp_l. Let $\bar{p}_l = p_l - \alpha p_l$, then $\bar{p}_l$ can be interpreted as the maximum amount that the carrier is willing to pay for serving the request or willingness-to-pay for short. Based on this observation, we can formulate the outsourcing requests selection problem for each agent as a mixed integer programming, in which the total revenue term in its objective function is replaced by the total maximal expenditure term. The notations used in the model are given as follows.

Indices

i, j, m: node index, where i, j, $m = 1,\ldots,N$ and N represents the number of nodes in the transportation network. The nodes include all pickup and delivery locations and all vehicle depots of the carriers

l: request index

Parameters

R_a the set of requests collected by the carrier from its shippers and at the state of *before execution*

R_b the set of requests acquired by the carrier from other carriers and at the state of *before execution*

P_i the set of requests whose pickup location is node i, $P_i \subseteq R_a \cup R_b$

D_i the set of requests whose delivery location is node i, $D_i \subseteq R_a \cup R_b$

d_l pickup and delivery quantity of request l, $l \in R_a \cup R_b$

α the minimum profit margin that the carrier wants to earn for serving a request

p_l price paid by a shipper to serve request l, $l \in R_a$

$\bar{p}_l$ the willingness-to-pay of the carrier for request l, $l \in R_a$, $\bar{p}_l = p_l - \alpha \cdot p_l$

p_l^o the outsourcing price of request l set by other carriers, which is the revenue obtained by the carrier to serve request l, $l \in R_b$

$\overline{p}_l^o$ the willingness-to-pay of the carrier for request l, $l \in R_b$, $\overline{p}_l^o = p_l^o - \alpha \cdot p_l^o$

W the number of vehicles of the carrier

C vehicle capacity of the carrier

o the depot node of the carrier

c_{ij} transportation cost from node i to j for each vehicle of the carrier, where $c_{ij} = c_{ji}$ and the triangle inequality $c_{im} + c_{mj} \geq c_{ij}$, holds for any i, j, m with $m \neq i$, $m \neq j$

τ_{ij} the traveling time from node i to node j for each vehicle of the carrier

a_i the earliest pickup/delivery time at node i

b_i the latest pickup/delivery time at node i

T_{ij} a large number, $T_{ij} = b_j - a_i$

Note that each time window is usually associated with the pickup or delivery operation of a request; in this paper, we consider the situation where each node involves at most one request, that is, the node is either the pickup location or the delivery location of a single request. For the situation where some nodes involve multiple requests, we can replace each of the nodes by a number of duplicate nodes where each duplicate node involves a single request, and the transportation costs among these duplicate nodes are set to zero. In this way, we can assume that each node in the transportation network involves only one request.

Variables

q_{ij} quantity of freight transported through arc (i, j)

x_{ij} the number of times that arc (i, j) is visited by vehicles of the carrier

y_l binary variable, $y_l = 1$ if request $l \in R_a$ is served by the carrier itself, $y_l = 0$ otherwise; $y_l = 1$ if request $l \in R_b$

t_i the time at which a vehicle of the carrier leaves node i. Since only one request involves the node, which implies that only one vehicle visits the node, t_i is well defined

With the notations, the mixed integer programming model of the outsourcing requests selection problem (ORSP) for the carrier is given as (1–13). Model ORSP:

$$Z = \mathrm{Max}\left(\sum_{l \in R_a} \bar{p}_l \cdot y_l + \sum_{l \in R_b} \bar{p}_l^o - \sum_{i=1}^{N} \sum_{j=1, j \neq i}^{N} c_{ij} \cdot x_{ij} \right) \tag{1}$$

Subject to:

$$\sum_{j=1, j \neq i}^{N} x_{ij} = \sum_{j=1, j \neq i}^{N} x_{ji}, \quad i = 1, \ldots, N, \tag{2}$$

$$q_{ij} \leq C \cdot x_{ij}, \quad i, j = 1, \ldots, N,\ i \neq j, \tag{3}$$

$$\sum_{j=1, j \neq i}^{N} q_{ij} - \sum_{j=1, j \neq i}^{N} q_{ji} = \sum_{l \in P_i} d_l \cdot y_l - \sum_{l \in D_i} d_l \cdot y_l, \quad i = 1, \ldots, N,\ i \neq o, \tag{4}$$

$$\sum_{j=1, j \neq o}^{N} q_{oj} = \sum_{l \in P_o} d_l \cdot y_l - \sum_{l \in D_o} d_l \cdot y_l, \tag{5}$$

$$\sum_{j=1, j \neq o}^{N} q_{jo} = 0, \tag{6}$$

$$\sum_{j=1}^{N} x_{oj} \leq W, \tag{7}$$

$$\sum_{j=1, j \neq i}^{N} x_{ij} \leq 1, \quad i = 1, \ldots, N,\ i \neq o, \tag{8}$$

$$t_j \geq t_i + \tau_{ij} \cdot x_{ij} - T_{ij} \cdot (1 - x_{ij}), \quad i, j = 1, \ldots, N,\ j \neq o,\ i \neq j, \tag{9}$$

$$x_{ij} \geq 0, x_{ij} \in Z, \quad i, j = 1, \ldots, N,\ i \neq j, \tag{10}$$

$$y_l \in \{0, 1\},\ l \in R_a \text{ and } y_l = 1,\ l \in R_b, \tag{11}$$

$$q_{ij} \geq 0, q_{ij} \in R, \quad i, j = 1, \ldots, N,\ i \neq j, \tag{12}$$

$$0 \leq a_i \leq t_i \leq b_i, \quad i = 1, \ldots, N, \tag{13}$$

The objective function (1) represents the surplus profit of the carrier obtained from serving requests, which includes three terms. The first term denotes the maximal expenditure of the carrier for fulfilling the requests collected from its shippers and selected (accepted) by it; the second term is a constant term representing the maximal expenditure of the carrier for fulfilling its acquired outsourcing requests; the third term represents the total transportation cost of the carrier for fulfilling the above-mentioned two sets of requests. Constraints (2) ensure that the number of vehicles leaving from a node (including the depot node) is equal to the number of vehicles arriving at the node. Constraints (3) are the vehicle capacity constraints. Constraints (4) are the flow conservation equations, assuring the flow balance at each node (except for the depot node). Constraints (5) and (6) are the flow conservation equations at the depot node, which insure that only empty vehicle is returned to the depot. Constraints (7) guarantee that the number of vehicles used by the carrier is at most equal to W. Constraints (8) ensures that each customer node is visited at most once by vehicles of the carrier. Constraints (9) indicate the relationship between the departure times of a vehicle from any two nodes. The constraints also assure that each vehicle of the carrier starts with and ends at its depot. Constraints (13) are the time window constraints for pickup/delivery operations at all nodes. Note that the objective function (1) of model ORSP is not the profit of the carrier, it is the part of the profit exceeding the expected profit $\sum_{l \in R_a} p_l \cdot y_l \cdot \alpha + \sum_{l \in R_b} p_l^o \cdot \alpha$. That is, when we calculate the profit of the carrier from the objective function value of the model, we should add the expected profit to the value. That is why we call the objective function the surplus profit of the carrier. The objective of the problem is to maximize the surplus profit.

Note that in the aforementioned model, we assume each carrier cannot outsource the requests acquired from other carriers, but it may outsource any request collected from its shippers with the state of *before execution*, i.e., it may choose to outsource an unexecuted request of its shippers later even if the request is chosen as a self-execution request currently. Each carrier solves its ORSP when it receives new requests from its shippers or it acquires requests from other carriers. Furthermore, a solution of the aforementioned model does not directly provide vehicle tours for fulfilling all selected requests (requests with $y_l = 1$), but the vehicle tours can be constructed easily from the solution since each node except for the depot node is visited at most once by a vehicle of the carrier.

Initial outsourcing price setting for outsourcing requests of each carrier

After the outsourcing requests selection process, each carrier must set an initial outsourcing price for each of its outsourcing requests. Since the objective function (1) of model ORSP can be interpreted as the profit surplus gained by the carrier for serving requests, the carrier naturally wants to maximize the profit surplus. If in an optimal solution of the model, $y_l = 1$, $l \in R_a$ implies that serving request l can increase the profit surplus; otherwise $y_l = 0$, $l \in R_a$ implies that serving request l will decrease the profit surplus. So the request l should be outsourced to other carriers. In this case, since serving request l by the carrier itself cannot achieve the expected profit αp_l, the initial outsourcing price of the request l, denoted by p_l^o, should be set to in the interval [0, $\bar{p}_l$], where $\overline{p_l} = p_l \cdot (1 - \alpha)$. Thus, the carrier can gain at least the expected (minimal) profit for request l if there is another carrier who accepts the outsourcing price for serving the request. According to this outsourcing price-setting method, the higher the minimum profit margin of the carrier, the lower the initial outsourcing price of the request.

5 Requests bidding problem

In our proposed multi-agent and auction-based framework, once the outsourcing requests are selected and priced by each agent, they will be added to an outsourcing request pool that gathers all outsourcing requests. All the requests are available to every agent. Each carrier can then determine whether it bids some requests in the pool as long as it is not empty, this decision problem is referred to as the requests bidding problem (RBP) or the requests acquisition problem that is formulated as a mixed integer programming (MIP) model in this section, which is similar to that of ORSP. The only difference between the two models is that some notations (parameters and variables) have different meanings. The notations with different meanings and used in the model of RBP are given as follows.

Parameters

R_a the set of requests accepted by the carrier for its self-execution at the state of *before execution*

R_{as} the set of requests collected from shippers and accepted by the carrier for its self-execution at the state of *before execution*

R_{ac} the set of requests acquired from other carriers and accepted by the carrier for its self-execution at the state of *before execution*; obviously, $R_a = R_{as} \cup R_{ac}$

R_b the set of requests outsourced by other carriers and at the state of *in auction*

Variables

y_l binary variable, $y_l = 1$ if request $l \in R_a$; $y_l = 1$ if $l \in R_b$ is selected to be acquired by the carrier, $y_l = 0$ otherwise

With the above-mentioned notations and other notations introduced in Sect. 4, the mixed integer programming model of RBP is given as (14)–(15).

Model RBP:

$$Z^I = \mathrm{Max}\left(\sum_{l\in R_{as}} \bar{p}_l + \sum_{l\in R_{ac}} \bar{p}_l^o + \sum_{l\in R_b} \bar{p}_l^o \cdot y_l - \sum_{i=1}^{N} \sum_{j=1,j\neq i}^{N} c_{ij} \cdot x_{ij}\right) \tag{14}$$

Subject to constraints (2–10), (12), (13), and

$$y_l = 1,\ l \in R_a \text{ and } y_l \in \{0,1\},\ l \in R_b, \tag{15}$$

The objective function (14) represents the surplus profit of the carrier obtained from serving requests, which includes four terms. The first and second terms are constant terms representing the maximal expenditure of the carrier for fulfilling the requests collected from its shippers and the requests acquired from other carriers, respectively; the third term represents the willingness-to-pay of the carrier for its bidding outsourcing requests; the fourth term represents the total transportation cost of the carrier for fulfilling the above-mentioned three sets of requests. The meanings of constraints (2–10), (12), and (13) are the same as those of model ORSP. Note that for each carrier, its requests bidding problem considers its resources (vehicles) and its currently acquired requests at the same time. Similar to model ORSP, the objective function (14) of model RBP is not the profit of the carrier, but it is the part of the profit exceeding the expected profit $\sum_{l\in R_{as}} p_l \cdot \alpha + \sum_{l\in R_{ac}} p_l^o \cdot \alpha + \sum_{l\in R_b} p_l^o \cdot y_l \cdot \alpha$.

6 Multiple asynchronous auction processes

In the multi-agent and auction-based framework, multiple auction processes of multiple outsourcing requests can happen at the same time. These processes progress asynchronously and interact with each other through requests outsourcing/acquisition and the adjustment of the outsourcing prices by the carriers. The processes have three important features: (1) Each auction process is request oriented rather than agent oriented, that is, each auction process corresponds to a request; (2) Each auction process involves multiple agents (carriers); and (3) The interaction of multiple auction processes (corresponding to multiple requests) and its impact on the behaviors of the agents are realized by the resource (vehicle) requirements (occupation) of these requests, because if an agent bids for a request, it may not able to bid for another request as the fulfillment of the former request will occupy some resource (vehicle) of the agent. In our framework, each auction is a double auction [24]; each agent plays two roles, one is auctioneer and the other is bidder. When an agent has chosen to outsource a request by solving an outsourcing requests selection problem described by model ORSP, it will act as an auctioneer and initiate the auction of the request; each of the other agents, which acts as a bidder, determines whether to bid for the request by solving a requests bidding problem described by model RBP. During the auction of a request, when the situation of a bidder is changed because it receives new requests from its shippers or its environment is changed because other agents announce the auction of new requests or the outsourcing prices of the requests in auction are updated, the bidder can change its bidding decision on the request made in a previous round and determine whether it stays in, drops from, or resubmits a bid in the current round of the auction. For this reason, each agent solves its ORSP and RBP every time when its situation is changed. In the following, we introduce in detail the auction for an outsourcing request, including iterative nature of the auction process, outsourcing price adjustment in the process, and conflict resolution in assigning it to an agent.

Given an outsourcing request l from the outsourcing request pool, its initial outsourcing price, its auction starting time t_l, and a fixed time period T_l are set by the outsourcing agent (auctioneer), the auction of request l is initiated by the auctioneer at time t_l and advances in an iterative way. At the end of each round when T_l is elapsed, the auctioneer may face three situations for determining which agent request l should be allocated to.

Situation 1: no agent bids for request l. In this situation, the auctioneer increases the outsourcing price of the request to attract more agents' interests in the request and restart another round of the auction. The new price should be adjusted in the interval $[0, \bar{p}_l]$, which has been explained in Sect. 4. The agent can increase the price in a regular fashion, which is similar to what in a well-known single-good auction named Japanese auction [19], where the price is successively raised. The price update rule increases the price by a fixed small amount δ from the current price, which is introduced by Lavi and Nisan [17]. More recently, Biswas and Narahari [4] applied this rule in developing efficient iterative auction mechanisms for combinatorial exchanges. Their numerical experiments show that the choice of different δ leads to different results. In their study, the value of various bidding and asking prices lies in the intervals [0, 1]. Accordingly, in this paper, we set $\delta_l = \rho \cdot p_l^0$ for request l, where p_l^o is the outsourcing price of this request and ρ is an adjustable parameter between 0 and 1.

Situation 2: only one agent bids for request l. Thus, the request is allocated to the bidding agent.

Situation 3: more than one agent bids for request l. Then the auctioneer decreases the outsourcing price of request l. The price decrease can employ a regular fashion similar to what in a well-known single-good auction named Dutch auction [19], where the price is successively lowered in the interval $[0, \bar{p}_l]$ by applying the same rule as proposed in *Situation 1* with the same parameter δ.

During the auction, the outsourcing price increases if *Situation 1* appears and the outsourcing price decreases if *Situation 3* appears. If at the end of a round, the situation of the auction is changed from 1 to 3 or from 3 to 1, the outsourcing price is set back to its value at the last round, then the auctioneer starts a new round of auction with parameter δ for the price update halved, i.e., set δ to $\delta/2$. Note that since an agent may engage in multiple auctions at the same time, any request acquired by the agent from an auction may influence acquisition choices of the agent in other auctions. So even for the same outsourcing request, a bidding agent may take different acquisition decisions at different moments.

The auction process is iteratively conducted by the auctioneer until one of the following two stopping conditions is satisfied: (1) Only one agent bids for request l and the request is allocated to the agent; (2) A given number of rounds is achieved or δ is small than a given number, if no agent bids for request l even with the price $\bar{p}_l$, then it is returned to the outsourcing agent; if multiple bidding agents compete for request l, a conflict resolution procedure is employed to determine which agent wins the request. The procedure allocates the request to the bidding agent who announces its bid earlier than others. Once an agent is awarded request l, this allocation is irreversible and the auction is terminated.

7 Simulation evaluation and comparison

To evaluate the performance of our multi-agent and auction-based framework for carrier collaboration, we compare it with a centralized framework (introduced in the "Appendix") proposed in the literature and the case of no collaboration among carriers by simulation. Twenty instances are randomly generated and simulated to evaluate the effectiveness of our proposed framework and approach.

7.1 Simulation design

In the simulation, we consider the following situation: all requests from different carriers are served on the same day, and they are not executed in advance until the collaborative transportation planning among all carriers is done. For simplicity, for each carrier, the time period for each round of auction is the same for all its outsourcing requests and all these requests with state *before auctio*n are auctioned simultaneously. However, different carriers may start their auctions at different times. For the first ten instances, each carrier has three requests and no other requests arrive during its auction processes, which means its outsourcing requests selection is performed only at the beginning; whereas for the second ten instances, each carrier only has two requests initially and the third request arrives at a random time in the auction processes. During the auction processes, each agent will not consider any of its outsourcing requests once they are auctioned, and when an auction is over with no other agents acquire a request, it will be returned to the auctioneer for reconsideration. An agent can organize multiple auctions and bid for multiple outsourcing requests at the same time.

The simulation is separated from the resolution of two decision problems (outsourcing requests selection problem and requests bidding problem) and other decision problems in the multi-agent and auction-based framework. This means that the resolution of these problems is assumed to take no time or take very short time. A system clock is introduced in the simulation. When a decision problem needs to be solved, we freeze the clock until the problem is solved. This can be regarded as that each decision problem is solved off-line. In addition, the simulation is driven by discrete events. The time is advanced in a discrete way. That is, we always advance the time to the moment at which the next event(s) will happen. After an event (or multiple events) happen (simultaneously), we update the system's state and determine the next event(s) and its (their) occurrence time.

Multiple events may occur in the simulation: the starting and the end of the auction process of a request, the arrival of a new request; the starting and end of a round of the auction of a request, etc. To implement the simulation, we define the state space of the multi-agent system and all events that may happen in the system, where the state of the system is represented by the state of each agent that is defined in turn by the states of its requests and the availability profile of its resources (vehicles). In the simulation, we assume that the auction of each outsourcing request starts immediately after the outsourcing carrier solves its ORSP problem. When a request is not allocated to any carrier after the termination of its auction, it is reconsidered by the outsourcing carrier as a new arrival request. All new arrival requests of each carrier will be only considered when it solves its ORSP problem. To help us understand the simulation, we first explain it by an illustrative example in the next subsection.

7.2 Demonstration of the simulation by an illustrative example

We first use the illustrated example in Sect. 2 to carry on the simulation process. The parameters of the example are explained as follows. R_a, R_b, and R_c denote the self-executed requests set for carriers a, b, c, respectively, which includes the requests collected from shippers and the requests acquired from other carriers. R_{ab}, R_{bb}, and R_{cb} denote the set of requests bid for by carriers a, b, c, respectively. R_o and R_{ao}, R_{bo}, R_{co} denote the set of outsourcing requests for all carriers and carriers a, b, c, respectively. P_a, P_b, and P_c denote the profit gained from fulfilling self-executed set and bided set for carriers a, b, c, respectively. P_{ao}, P_{bo}, and P_{co} denote the profit gained by carriers a, b, c from outsourcing their requests, respectively. The minimum profit margin of each carrier, α, is set to 5%. p_l denotes the price paid by a shipper for request l; p_l^o denotes the outsourcing price of request l, which is set to $\bar{p}_l$ initially and is adjusted in the interval $[0, \bar{p}_l]$, where $\overline{p_l} = p_l \cdot (1 - \alpha)$. The parameter ρ of the outsourcing price update rule is set to 0.1 for all carriers, and δ_l in the price update rule is set to $p_l^o \cdot \rho$ request l. The time period for each round of auction for each outsourcing request of carriers a, b, c is set to 5, 10, 15 s, respectively. Carriers a, b, c has two requests initially, the arrival time of the third request is set to 4, 8, 13 s, respectively. Carriers a, b, c enter the actions at 1, 2, 3 s, respectively.

Besides, some abbreviations and symbols are defined as follows. ORSP*i* represents the outsourcing requests selection process for carrier i. RAP*ilk* denotes round k of the auction process of outsourcing request l of carrier i. RBP*ilk* denotes round k of bidding process of request l for carrier i. RLA*lik* denotes that request l is allocated to carrier i in round k of its auction. All involved decision problems (models) are solved directly by using IBM MIP Solver ILOG Cplex 12. The events happened in all the auction processes for each carrier are listed in Table 3.

The profit comparison is given in Table 4, where *IP* denotes the initial profit of each carrier before collaboration calculated by model *ORSP*; *PMAA* denotes the profit obtained by our multi-agent and auction-based framework; *PC* denotes the profit obtained by the centralized framework presented in the appendix, in which all collected requests from carriers are allocated in a globally optimal way.

From Table 4, we can see that *PMAA* can generate a better requests allocation solution compared with *IP*. *PMAA* fulfills all requests as well as *PC*. Each carrier's profit increases in *PMAA*. This example demonstrates the effectiveness and feasibility of our proposed approach.

7.3 Simulation results on randomly generated instances

To evaluate the performance of our framework, we randomly generate two sets of instances, where each set has ten instances. The parameters of the instances are taken as follows. The transportation network for each instance has 21 nodes ($N = 21$). The coordinates of each node are randomly and uniformly generated from 42 × 42 square; as soon as the coordinates of all nodes are generated, the Euclidean distance between any two nodes i and j, denoted by d_{ij}, is calculated. For simplicity but without loss of generality, we set $c_{ij} = \tau_{ij} = d_{ij}$ for all carriers. There are three carriers ($K = 3$) operated in the transportation network. Every carrier randomly selects a node as its depot node, but different carriers have different depot nodes; each carrier owns a random number of vehicles chosen between

Table 3 Discrete event lists for the illustrative example

Time	Events of carrier a	Events of carrier b	Events of carrier c
1	ORSPa, $R_a = \{r1\}$ RAP121, $R_{ao} = \{r2\}$ $R_o = \{r2\}$, $p_2^o = 66.5$ $R_{ab} = \{\}, P_{ao} = 0$, $P_a = 36.8$	–	–
2	–	ORSPb, $R_b = \{r4\}$ RAP251, $R_o = \{r2, r5\}$ $R_{bo} = \{r5\}$, $p_5^o = 59.85$ RBP221, $R_{bb} = \{r2\}$ $P_{bo} = 0$, $P_b = 81.3$	–
3	–	–	ORSPc, $R_c = \{r7,r8\}$ RBP321, RBP351 $R_{cb} = \{r2\}$, $P_c = 178.6$
4	ORSPa, $R_a = \{r1,r3\}$ RBP151, $R_{ab} = \{r5\}$, $P_a = 200.75$	–	–
6	RAP122,$p_2^o = 59.85$	–	–
7	–	RBP222 $R_{bb} = \{\}$, $P_b = 75.6$	RBP322, RBP351 $R_{cb} = \{\}$, $P_c = 171$
8	–	ORSPb, $R_b = \{r4,r6\}$ RBP222, $R_{bb} = \{r2\}$, $P_b = 118.95$	–
11	RLA222, $R_o = \{r5\}$ $P_{ao} = p_2 - p_2^o = 70 - 59.85$	ORSPb, $R_b = \{r4,r6,r2\}$ $P_b = 118.95$	–
12	ORSPa, $R_a = \{r1,r3,r5\}$ $P_a = 200.75$	RLA511, $R_o = \{\}$ $P_{bo} = p_5 - p_5^o = 63 - 59.85$	–
13	–	–	ORSPc, $R_c = \{r7, r9\}$ RAP381,$R_{co} = \{r8\}$ $R_o = \{r8\}, P_{co} = 0$ $p_8^o = 49.4, P_c = 182.4$
14	RBP181 $R_{ab} = \{r8\}$, $P_a = 217.35$	RBP281 $R_{bb} = \{r8\}$, $P_b = 158.45$	–
28	–	–	RAP382,$p_8^o = 44.46$
29	RBP182 $R_{ab} = \{r8\}$, $P_a = 212.41$	RBP282 $R_{bb} = \{r8\}$, $P_b = 153.51$	–
43	–	–	RAP383,$p_8^o = 39.52$
44	RBP183 $R_{ab} = \{r8\}$, $P_a = 207.47$	RBP283 $R_{bb} = \{r8\}$, $P_b = 148.57$	–
58	–	–	RAP384,$p_8^o = 34.58$
59	RBP184 $R_{ab} = \{r8\}$, $P_a = 202.53$	RBP284 $R_{bb} = \{r8\}$, $P_b = 143.63$	–
73	–	–	RAP385,$p_8^o = 29.64$
74	RBP185 $R_{ab} = \{\}$, $P_a = 200.75$	RBP285 $R_{bb} = \{r8\}$, $P_b = 138.69$	–
88	–	ORSPb, $P_b = 138.69$ $R_{bb} = \{r4, r6, r2, r8\}$	RLA825, $R_o = \{\}$ $P_{co} = p_8 - p_8^o = 52 - 29.64$
	Total profit is $P_a + P_{ao} = 210.9$	Total profit is $P_b + P_{bo} = 141.84$	Total profit is $P_c + P_{co} = 204.76$

1 and 10, and each vehicle has the same capacity $C = 10$. The number of transportation requests L is set to $(N - K)/2$ $(L = 9)$. Carrier a owns request 1, 2, 3, and carrier b owns request 4, 5, 6, and carrier c owns request 7, 8, 9. The requests are generated by randomly choosing a pickup node i (not depot node) and a delivery node j ($j \neq i$, not

Table 4 Profit comparison between three cases for the illustrative example

Profit without collaboration			Profit with collaboration					
IP			PMAA			PC		
Carrier a	Carrier b	Carrier c	Carrier a	Carrier b	Carrier c	684.4		
146	97.7	182.4	210.9	141.84	204.76			
426.1 in total			557.5 in total					
Fulfilled requests								
Carrier a	Carrier b	Carrier c	Carrier a	Carrier b	Carrier c	Carrier a	Carrier b	Carrier c
r1, r3	*r4, r6*	*r7, r9*	*r1, r3, r5*	*r2, r4, r6, r8*	*r7, r9*	-	*r4, r8*	*r1, r2, r3, r5, r6, r7, r9*

Table 5 The time parameters of the twenty instances

Instance	1	2	3	4	5	6	7	8	9	10
Time period (s)	–	15	10	15	–	–	15	10	5	5
	10	5	–	5	5	10	10	–	10	–
	5	–	15	10	10	5	5	5	15	10
Instance	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**	**19**	**20**
Time period (s)	5	–	10	10	5	15	–	10	15	5
	15	–	5	15	10	5	15	–	10	15
	10	5	15	5	–	–	10	–	5	10
New request arrival time (s)	6	10	15	10	15	13	16	11	4	4
	9	5	20	5	18	15	3	5	19	9
	10	16	10	15	13	7	10	18	6	13

depot node), each node is just chosen once; each request is assigned with a random quantity of freight d that is not larger than a predefined number. For the first set of instances, this number is set to five for instances 1–5 and set to 10 for instances 6–10. For the second set of instances, this number is set to five for instances 11–15 and set to 10 for instances 16–20. The ask price paid by a shipper to a carrier for serving a request is defined by $\gamma* (1 + \alpha)* \beta* (c_{oi} + c_{ij} + c_{oj})$, where o denotes the depot of the carrier and $c_{oi} + c_{ij} + c_{oj}$ is the direct shipping cost of the request, β is the average utilization of vehicle in ideal situations for serving the request, α denotes the minimum profit margin of the carrier, γ is an adjustable parameter no less than one. The calculation of β is explained through a simple example as follows: assume that the capacity of each vehicle of the carrier is 10 and it has three requests to serve, whose pickup/delivery quantities are 3, 5, and 8, respectively, and the considered request is the request with quantity five, so the total quantity of the requests is $(3 + 5 + 8) = 16$ and the carrier requires at least two vehicles to serve all the requests, β is then calculated as 2*5/16 for the second request. Parameter α is set to 5% for all carriers according to Holguín-Veras et al. [12]. Parameter γ is set to two, which ensures that the ask price of the request is sufficiently larger than its service cost in order to avoid the situation that no carriers choose to serve the request. The time windows are generated in the following way: the time interval for serving all requests is set to [0, 240] (480 min = 8 h, time unit is taken as 2 min); the earliest service time a_i at pickup node i is randomly chosen in $[\tau_{oi}, 240\text{-}\tau_{oj} - \tau_{ij}]$, where τ_{oi} is calculated by considering the farthest depot o (three depots in total) from node i; the latest service time b_i is randomly chosen from $(a_i + 15)$ to $(a_i + 45)$; the earliest service time a_j at delivery node j is randomly chosen in $[\tau_{oi} + \tau_{ij}, 240\text{-}\tau_{oj}]$; and latest service time b_j is randomly chosen from $(a_j + 15)$ to $(a_j + 45)$. And the conditions $(b_j - a_i) > c_{ij}$, $b_i \leq 240$ and $b_j \leq 240$ are verified during the time windows generation process. The time windows for each carrier's depot are set to [0, 240]. The parameter ρ in the outsourcing price update rule is set to 0.1 for all carriers.

The other parameters are given in Table 5. For each column (instance), the three (sub)rows in row "*Time period*" denote the time period for each round of auction for three carriers, respectively, which is randomly chosen from three numbers (5, 10, 15), and it is vacant if a carrier does not outsource requests. For the second set of instances, each carrier has two requests initially, the three (sub)rows

Table 6 Results comparison of the first set of instances

Instance	1	2	3	4	5
IP					
Carrier a	75.3	15.4	41.3	10.2	36.5
Carrier b	36	12.7	155	133.9	105.6
Carrier c	2	167.8	19	96.9	19.8
Total	113.3	195.9	215.3	241	161.9
Fulfilled requests	*r1*, *r2*, *r3*, *r5*, *r6*, *r7*	*r3*, *r6*, *r7*, *r8*, *r9*	*r1*, *r3*, *r4*, *r5*, *r6*, *r8*	*r1*, *r3*, *r5*, *r6*, *r7*, *r8*	All except *r5*, *r7*
PMAA					
Carrier a	75.3	15.4	46.955	28.96	45.34
Carrier b	59.35	36.82	155	200.59	126.96
Carrier c	3.35	200.48	26.145	99.2	19.8
Total	138	252.7	228.1	328.75	192.1
Fulfilled requests	All except *r4*, *r8*	All except *r1*, *r2*	All except *r7*, *r9*	All except *r2*	All except *r7*
PC					
Total	184.3	346.8	366.6	458.4	340.6
Fulfilled requests	All except *r4*, *r8*	All	*r1*, *r3*, *r4*, *r5*, *r6*, *r8*	All except *r2*	All

Instance	6	7	8	9	10
IP					
Carrier a	184.2	150.1	148.4	92.9	73.1
Carrier b	60.4	94.6	93.7	173.5	160.4
Carrier c	135.6	183.1	120.6	139.8	153.9
Total	380.2	427.8	362.7	406.2	387.4
Fulfilled requests	*r1*, *r2*, *r3*, *r4*, *r7*	*r1*, *r2*, *r4*, *r5*, *r8*, *r9*	*r2*, *r3*, *r4*, *r5*, *r6*, *r7*	*r1*, *r3*, *r4*, *r6*, *r7*, *r8*	*r1*, *r4*, *r5*, *r6*, *r8*, *r9*
PMAA					
Carrier a	287.575	154.4	189.945	92.9	77.45
Carrier b	85.725	118.8	116.7	175.1	227.15
Carrier c	140	183.1	151.855	143.6	157.7
Total	513.3	456.3	458.8	411.6	462.3
Fulfilled requests	All	All except *r7*	All	All except *r2*, *r9*	All except *r2*
PC					
Total	615	591.3	607.8	450.7	519.3
Fulfilled requests	All	All except *r6*	All	All except *r2*, *r9*	All except *r2*

in row "*New request arrival time*" denote the arrival time of a new request for each carrier; in each instance, it is generated randomly in the intervals [1, 20].

All decision problems (models) involved in these instances are solved by using ILOG Cplex 12 directly. All simulation results are given in Tables 6 and 7, where *IP*, *PMAA*, and *PC* have the same meaning as in Table 4 in last subsection. Both of the rows *IP* and *PMAA* have five (sub)rows, the first four (sub)rows indicate the profit of carriers *a*, *b*, *c*, and the total profit of the three carriers, respectively, the last (sub)row indicates the fulfilled requests. The row *PC* has only two (sub) rows, which indicate the total profit and the fulfilled requests, respectively.

From the results, we can observe that our multi-agent and auction-based framework (row *PMAA*) can obtain a

Table 7 Results comparison of the second set of instances

Instance	11	12	13	14	15
IP					
Carrier a	123.7	231.4	10.6	144.7	21.4
Carrier b	145.3	119.6	0	192.7	170.7
Carrier c	10.9	88.4	116.9	2	102.6
Total	279.9	439.4	127.5	339.4	294.7
Fulfilled requests	All except *r6*, *r9*	All except *r7*	*r2*, *r7*, *r9*	*r1*, *r3*, *r4*, *r6*, *r9*	All except *r3*, *r6*
PMAA					
Carrier a	131.75	231.4	14.4425	147.6	22.95
Carrier b	148.35	136.375	13.6075	194.95	172.8
Carrier c	26.2	100.125	158.95	60.55	116.05
Total	306.3	467.9	186.98	403.1	311.8
Fulfilled requests	All except *r9*	All	*r2*, *r5*, *r6*, *r7*, *r9*	All except *r7*, *r8*	All
PC					
Total	393.5	551.4	243.9	534.3	496.4
Fulfilled requests	All except *r9*	All	All except *r1*, *r4*	All	All

Instance	16	17	18	19	20
IP					
Carrier a	192.5	124.8	139.9	111	92
Carrier b	99.1	180.4	134.8	176.3	32.5
Carrier c	146.4	119.8	116.6	142.7	55.5
Total	438	425	391.3	430	180
Fulfilled requests	All except *r3*, *r6*	All except *r6*, *r7*	All except *r1*, *r8*	*r1*, *r2*, *r4*, *r5*, *r7*, *r8*	All except *r1*, *r5*
PMAA					
Carrier a	192.5	124.8	149.005	132.4	100.45
Carrier b	102.55	200.63	134.8	179.6	55.65
Carrier c	169.45	141.87	139.595	142.7	55.5
Total	464.5	467.3	423.4	454.7	211.6
Fulfilled requests	All except *r3*	All	All except *r1*, *r8*	All except *r3*, *r9*	All
PC					
Total	587.6	622.2	561.7	511.4	212
Fulfilled requests	All	All	All	All except *r9*	All

larger profit in total than the profit gained in the case without collaboration among carriers (row *IP*) in all instances. *PMAA* can fulfill more requests than *IP*. The profit of each carrier is not decreased in *PMAA* compared with its profit in *IP*. Most of them can increase their profits. The centralized framework (row *PC*) leads to better results than *PMAA*, which is probably because of two reasons. One is that *PC* maximizes the total profit of the whole alliance by sharing all information of the carriers, and it is based on global optimization; the other is that some high profitable requests for a carrier may be already acquired by other carriers in *PMAA*. However, *PC* needs a fair and feasible post-coordination profit allocation mechanism, although it is not easy to measure the contribution of each carrier in the alliance. Furthermore, *PMAA* is more acceptable by carriers since no private information of each carrier is revealed to others and no profit reallocation mechanism is needed. *PMAA* is more flexible to deal with dynamic market situations, where the time for the auction and acquisition of each request is uncertain in advance. Therefore, *PMAA* is more realistic, promising, and easier to be applied in practice.

8 Conclusion

A carrier collaboration problem in less than truckload transportation with pickup and delivery requests (CCPLTL) has been studied in this study. A multi-agent and auction-based framework is proposed for the problem together with relative decision models. The decentralization is the major feature of the framework, where each carrier acts both as an auctioneer and as a bidder, no private information is shared among carriers, and no post-coordination profit reallocation among the carriers is required. Two key decision issues for each carrier in the framework are addressed: one is outsourcing requests selection problem and the other is outsourcing requests bidding problem. These two problems are formulated mathematically. Twenty instances are randomly generated and simulated to evaluate the performance of our framework compared with the case without collaboration among carriers and with a centralized framework. The results show that our approach can increase the profit of each carrier and the total profit of all carriers in most instances through reducing transportation costs. In our future work, we will try to extend our proposed framework to deal with the outsourcing and acquisition of bundles of transportation requests.

Appendix: centralized framework for carrier collaboration

Under the centralized framework, multiple carriers come to an agreement for cooperating each other and constitute a collaborative alliance with a coordinator in charge of making collaborative transportation plans for the alliance. The coordinator may be a virtual one. The objective of the coordinator is to maximize the total profit of the alliance, which will then be allocated among the carriers. Regarding the problem investigated in this paper, the coordinator in the centralized framework determines the reallocation of all transportation requests to the carriers so as to maximize the total profit. This centralized requests reallocation problem (CRRP) for CCPLTL can be formulated as a mixed integer programming (MIP). The notations used in the MIP model include the notations introduced in Sect. 4 and the following notations.

Indices

k: carrier index, $k = 1, \ldots, K$, where K represents the number of carriers

l: request index, $l = 1, \ldots, L$, where L represents the total number of requests of all carriers

Parameters

C_k vehicle capacity of carrier k

W_k the number of vehicles owned by carrier k

o_k the depot of carrier k

α_k the minimum profit margin of carrier k

c_{ij}^k transportation cost from node i *to* j for each vehicle of carrier k

τ_{ij}^k the traveling time from node i to node j for each vehicle of the carrier k

Variables

q_{ij}^k quantity of freight transported through arc (i, j) by vehicles of carrier k

x_{ij}^k the number of times that arc (i, j) is visited by vehicles of carrier k

y_{lk} 1 if request l is reallocated to carrier k, 0 otherwise

t_i^k the time at which a vehicle of carrier k leaves node i

With the notations, the mixed integer programming model is given as (16–29).

Model CRRP:

$$Z^{II} = \mathrm{Max}\left(\sum_{k=1}^{K}\sum_{l=1}^{L} p_l \cdot y_{lk} \cdot (1-\alpha_k) - \sum_{k=1}^{K}\sum_{i=1}^{N}\sum_{j=1, j\neq i}^{N} c_{ij}^k \cdot x_{ij}^k\right) \tag{16}$$

Subject to:

$$\sum_{j=1, j\neq i}^{N} x_{ij}^k = \sum_{j=1, j\neq i}^{N} x_{ji}^k, \quad i = 1, \ldots, N, \ k = 1, \ldots, K, \tag{17}$$

$$q_{ij}^k \leq C_k \cdot x_{ij}^k, \quad i,j = 1, \ldots, N, \ i \neq j, \ k = 1, \ldots, K, \tag{18}$$

$$\sum_{j=1, j\neq i}^{N} q_{ij}^k - \sum_{j=1, j\neq i}^{N} q_{ji}^k = \sum_{l\in P_i} d_l \cdot y_{lk} - \sum_{l\in D_i} d_l \cdot y_{lk}, \quad i = 1, \ldots, N, \ i \neq \{o_k\}, \ k = 1, \ldots, K, \tag{19}$$

$$\sum_{j=1, j\neq o_k}^{N} q_{o_k j}^k = \sum_{l\in P_{o_k}} d_l \cdot y_{lk} - \sum_{l\in D_{o_k}} d_l \cdot y_{lk}, \quad k = 1, \ldots, K, \tag{20}$$

$$\sum_{j=1, j\neq o_k}^{N} q_{j o_k}^k = 0, \quad k = 1, \ldots, K, \tag{21}$$

$$\sum_{j=1,j\neq o_k}^{N} x_{o_k j}^k \leq W_k, \quad k=1,\ldots,K, \tag{22}$$

$$\sum_{k=1}^{K} y_{lk} \leq 1, \quad l=1,\ldots,L, \tag{23}$$

$$\sum_{j=1,j\neq i}^{N} x_{ij}^k \leq 1, \quad i=1,\ldots,N,\ i\neq\{o_k\},\ k=1,\ldots,K, \tag{24}$$

$$\begin{aligned} & t_j^k \geq t_i^k + \tau_{ij}^k \cdot x_{ij}^k - T_{ij}\cdot(1-x_{ij}^k), \\ & \quad i,j=1,\ldots,N,\ j\neq o_k,\ i\neq j,\ k=1,\ldots,K, \end{aligned} \tag{25}$$

$$x_{ij}^k \geq 0, x_{ij}^k \in Z, \quad i,j=1,\ldots,N, \quad i\neq j,\ k=1,\ldots,K, \tag{26}$$

$$y_{lk} \in \{0,1\}, \quad l=1,\ldots,L,\ k=1,\ldots,K, \tag{27}$$

$$q_{ij}^k \geq 0, q_{ij}^k \in R, \quad k=1,\ldots,K,\ i,j=1,\ldots,N,\ i\neq j, \tag{28}$$

$$0\leq a_i \leq t_i^k \leq b_i, \quad i=1,\ldots,N,\ i\neq o_k,\ k=1,\ldots,K, \tag{29}$$

The objective function (16) represents the surplus profit for the alliance by serving all selected requests (requests with the sum of $y_{lk}=1,\ k=1,\ldots,K$). Constraints (17) ensure that the number of vehicles of each carrier leaving from each node (including depot nodes) is equal to the number of vehicles of the carrier arriving at the node. Constraints (18) are the vehicle capacity constraints for each carrier. Constraints (19) are the freight flow conservation equations for each carrier, assuring the flow balance at each node (except for all depot nodes) for each carrier. Constraints (20) and (21) are the flow conservation equations on the depot node of each carrier, which insure that only empty vehicle is returned to the depot. Constraints (22) imply that no more than W_k vehicles can be used for carrier k. Constraints (23) guarantee that each request is allocated to at most one carrier. Constraints (24) ensure that each node (except for depot nodes) can be visited at most once by vehicles of each carrier. Constraints (25) indicate the time window constraints for pickup/delivery operations at all nodes for each carrier. The constraints also assure that every vehicle of each carrier starts with and ends at its depot. Constraints (29) are the time window constraints for pickup/delivery operations at all nodes for each carrier. Similar to the models presented in Sects. 4 and 5, the total profit of the alliance is the sum of the surplus profit obtained by solving model CRRP and the expected profit $\sum_{k=1}^{K}\sum_{l=1}^{L} p_l \cdot y_{lk} \cdot \alpha_k$.

References

1. Agarwal R, Ergun O (2008) Mechanism design for a multicommodity flow game in service network alliances. Oper Res Lett 36(5):520–524
2. Agnetis A, Pacciarelli P, Pacifici A (2007) Combinatorial models for multi-agent scheduling problems. *Chapter 2 of book multiprocessor of scheduling: theory and applications*. Itech Education and Publishing, Austria
3. Berger S, Bierwirth C (2010) Solutions to the request reassignment problem in collaborative carrier networks. Transp Res Part E 46(5):627–638
4. Biswas S, Narahari Y (2009) Approximately efficient iterative mechanisms for combinatorial exchanges. In: IEEE conference on commerce and enterprise computing, Vienna, Austria, pp 182–187
5. Bürckert HJ, Fischer K, Vierke G (1998) TeleTruck: a holonic fleet management system. In: Proceedings of the 14th European meeting on cybernetics and systems research, vol 2, Vienna, Austria, pp 695–700
6. Dai B, Chen HX (2009) Mathematical model and solution approach for collaborative logistics in less than truckload (LTL) transportation. In: 39th international conference on computers & industrial engineering, Troyes, France
7. Dai B, Chen HX (2009) A benders decomposition approach for collaborative logistics planning with LTL transportation. The third international conference on operations and supply chain management, Wuhan, China
8. Davidsson P, Henesy L, Ramstedt L, Törnquist J, Wernstedt F (2005) An analysis of agent-based approaches to transport logistics. Transp Res Part C 13:255–271
9. Ergun Ö, Kuyzu G, Savelsbergh M (2007) Shipper collaboration. Comput Oper Res 34(6):1551–1560
10. Ergun Ö, Kuyzu G, Savelsbergh M (2007) Reducing truckload transportation costs through collaboration. Transp Sci 41(2):206–221
11. Houghtalen L, Ergun Ö, Sokol J (2007) Designing allocation mechanisms for carrier alliances. Available at the website: http://www.agifors.org/award/submissions2007/Houghtalen_paper.pdf
12. Holguín-Veras J, Xu N, Jong G, Maurer H (2009) An experimental economics investigation of shipper-carrier interactions in the choice of mode and shipment size in freight transport. Netw Spatial Econ. doi:10.1007/s11067-009-9107-x
13. Krajewska MA, Kopfer H (2006) Collaborating freight forwarding enterprises: request allocation and profit sharing. OR Spectrum 28(2):301–317
14. Krajewska M, Kopfer H, Laporte G, Ropke S, Zaccour G (2008) Horizontal cooperation of freight carriers: request allocation and profit sharing. J Oper Res Soc 59:1483–1491
15. Kwon RH, Lee CG, Ma Z (2005) An integrated combinatorial auction mechanism for truckload transportation procurement, *Working paper*, Mechanical and Industrial Engineering, the University of Toronto, Canada
16. Lang NA, Moonen JM, Srour FJ, Zuidwijk RA (2008) Multi agent systems in logistics: a literature and state-of-the-art review. Available at the website: http://www.hdl.handle.net/1765/12902
17. Lavi R, Nisan N (2005) Online ascending auctions for gradually expiring items. In: Proceedings of the sixteenth annual ACM-SIAM symposium on Discrete algorithms, Vancouver, British Columbia, Canada, pp 1146–1155
18. Lee CG, Kwon RH, Ma Z (2007) A carrier's optimal bid generation problem in combinatorial auctions for transportation procurement. Transp Res Part E 43(2):173–191
19. McAfee RP, John M (1987) Auctions and bidding. J Econ Lit 25(2):699–738
20. Mes M, Van der heijden M (2007) Pricing and scheduling strategies for carriers and shippers in sequential transportation auctions. In: Proceedings of the sixth triennial symposium on transportation analysis, Phuket Island, Thailand
21. Mes M (2008) Sequential auctions for full truckload allocation. PhD thesis, University of Twente, Enschede, The Netherlands

22. Sandholm T (1993) An implementation of the contract net protocol based on marginal cost calculations. In: Proceedings of the 11th national conference on artificial intelligence, Washington, USA, pp 256–262
23. Schwind M, Gujo O, Vykoukal J (2009) A combinatorial intra-enterprise exchange for logistics services. Inf Syst e-Bus Manage 7(4):447–471
24. Shoham Y, Brown KL (2009) Multiagent systems algorithmic, game-theoretic, and logical foundations. Cambridge University Press, UK
25. Solistics solutions Inc. Irvine, California, USA. Freight Terms, available at the website: http://www.solistics.com/Documents/Freight%20Terms.pdf
26. Villahoz JJL, Martinez RO, Arauzo AA, Ordax JMG (2010) Price updating in combinatorial auctions for coordination of manufacturing multiagent systems. In: 8th International conference on practical applications of agents and multiagent systems. Springer, New York, pp 201–207
27. Vries S, Vohra RV (2003) Combinatorial auctions: a survey. INFORMS J Comput 15(3):284–309
28. Walsh WE, Wellman MP (2003) Decentralized supply chain formation: a market protocol and competitive equilibrium analysis. J Artif Intell Res 19:513–567
29. Wellman MP (1992) A general-equilibrium approach to distributed transportation planning. In: Proceedings of the 10th national conference on artificial intelligence, San Jose, USA, pp 282–289
30. Wellman MP, Walsh WE, Wurman PR, MacKie-Mason JK (2001) Auction protocols for decentralized scheduling. Games Econ Behav 35:271–303
31. Wooldridge M (2009) An introduction to multi agent systems—second edition. Wiley, USA