

Heterogeneous multi-depot collaborative vehicle routing problem

Qihuan Zhang^a, Ziteng Wang^b, Min Huang^{a,*}, Yang Yu^a, Shu-Cherng Fang^c

^a College of Information Science and Engineering; State Key Laboratory of Synthetical Automation for Process Industries, Northeastern University, Shenyang, Liaoning, 110819, China

^b Department of Industrial and Systems Engineering, Northern Illinois University, DeKalb, IL 60115, USA

^c Edward P. Fitts Department of Industrial and Systems Engineering, North Carolina State University, Raleigh, NC 27695, USA

ARTICLE INFO

Keywords:

Vehicle routing
Multiple heterogeneous depots
Collaborative distribution
Benders' decomposition
Combinatorial Benders' cuts

ABSTRACT

Collaborative vehicle routing of multiple logistics providers is an important component of horizontal logistic collaboration that generates economic and societal benefits. Existing research on collaborative vehicle routing is limited to the homogeneous setting where the logistics providers transport the same product. To better address the need of a general modeling framework and fast computational methods for the growth of collaboration among logistics providers carrying various products, we investigate a heterogeneous multi-depot collaborative vehicle routing problem (HMCVRP) in this paper. The key operational and computational challenge of realizing the collaborative route planning is to properly select transfer points for product transshipment between vehicles of different depots. We propose a Benders-based branch-and-cut algorithm with the technique of combinatorial Benders' cuts to solve a mixed-integer programming formulation of HMCVRP. Numerical experiments indicate that the proposed algorithm significantly outperforms the CPLEX solver using the commonly adopted big-M transformation-based method. Additional computational study further reveals the importance of the locations of depots and having a well-designed cost savings allocation mechanism in practice.

1. Introduction

The logistics industry plays an increasingly important role in global economy by fueling the consumption and driving the development of other sectors including manufacturing, agriculture and retail. Taking China as an example, the express delivery market and its associated industries contribute to 1% of the nation's GDP with approximately 7% of the domestic employment (Shen et al., 2020). In recent years, the logistics industry in China sees rising concentration as the market competition becomes extremely fierce and the resulting low profit margin is troubling many logistics providers, especially the small ones (Cruijssen et al., 2007b; Gansterer and Hartl, 2018). In particular, last-mile delivery accounts for about 28% of the total logistics distribution cost and becomes a bottleneck of the development of the logistics industry (Bergmann et al., 2020; Wang et al., 2016). Companies are actively seeking innovative strategies and solutions to the last-mile delivery problem. For instance, Alibaba launched the “Rookie Station” platform in 2013 that integrates order information of an alliance of logistics providers to simplify their route planning.

Collaboration in delivery and pickup operations, such as joint route planning and order exchange, can allow logistics providers to improve their operational efficiency and reduce transportation costs in last-mile delivery. They may utilize the synergies of the collaborative operations to achieve economies of scale. Different from the common vertical collaboration, in this scenario, companies form alliances horizontally with other companies in the same layer of the supply chain. In general, horizontal collaboration in supply

* Corresponding author.

E-mail address: mhuang@mail.neu.edu.cn (M. Huang).

chains can lead to benefits in economic, environmental and societal aspects. (For more extensive reviews of horizontal collaboration please refer to [Gansterer and Hartl 2018, 2020](#), [Pan et al. 2019](#) and [Verdonck et al. 2013](#)) While collaboration could be interpreted and implemented very broadly, the focus of this paper is on collaborative vehicle routing. In the rest of the paper, we use the word “collaboration” in this sense unless stated otherwise.

The benefit of collaborative route planning is prominent. [Gansterer and Hartl \(2018\)](#) and [Soysal et al. \(2018\)](#) state that collaboration outperforms non-cooperative setting by 20%–30% in cost. [Crujssen et al. \(2007a\)](#) systematically analyze the potential advantages of collaborative vehicle routing and indicate that 30% of “synergy values” (i.e., cost reduction) will be achieved through joint route planning. [Ergun et al. \(2007\)](#) study the cost savings of collaboration by shippers to reduce empty trucks reposition. [Agarwal and Ergun \(2008\)](#) consider the collaborative network design in liner shipping and show that carriers can increase the utilization of ships’ capacity through joint ship scheduling. [Mancini et al. \(2021b\)](#) introduce the time and service consistency and workload balance into a multi-period collaborative vehicle routing problem. Experiments show these factors will not bring additional cost for collaboration. [Lin \(2008\)](#) conducts experiments with real-life and simulated data to show the cost savings of the cooperative strategy in a vehicle routing problem with pickup and delivery time windows over the non-cooperative setting. [Sprenger and Mönch \(2012\)](#) investigate the cooperative transportation planning problem emerged from German food industry where multiple manufacturers share vehicle fleets in delivery services, and they clearly assess the collaboration potential in a dynamic and stochastic system. The economic benefit of collaborative route planning is also confirmed by [Liu et al. \(2010a\)](#), [Buijs et al. \(2016\)](#), among other studies. In addition to reducing transportation cost and providing competitive advantages for the participating companies, collaboration of logistics providers can bring positive impact to society and the environment ([Gansterer and Hartl, 2020](#)), such as reduction of road congestion, vehicle emission and noise pollution. For this reason, collaboration is also encouraged by public authorities. [Soysal et al. \(2018\)](#) propose a model for horizontal collaboration in inventory routing problem that can result in 8%–33% in emission.

The above studies have illustrated the advantages of collaboration from different perspectives via the centralized programming method, under the setting that the requests can be exchanged between collaborators. In this paper, we propose a different form of collaboration where any request must be met by the products of a specific provider, which means the direct exchanging of requests does not work.

Collaboration under incomplete information can also be properly managed to generate benefits. When companies are not willing to share full information, [Krajewska and Kopfer \(2006\)](#) propose an auction-based request exchange mechanism for carriers to collaborate. [Berger and Bierwirth \(2010\)](#) further divide the auction process into five stages to make the whole process clearer and easier to implement. Subsequently, [Gansterer and Hartl \(2016\)](#) propose a requests selection mechanism that takes the geographical factors into account. [Pekeć and Rothkopf \(2003\)](#) show that the combinatorial auction in collaborative vehicle routing problems can better realize the geographical synergy of requests and reduce the transportation cost. [Li et al. \(2016\)](#) consider the auction with reserved requests, i.e., these requests cannot be exchanged. To reduce the complexity caused by large number of potential request bundles, [Li et al. \(2015\)](#) propose an intuitive approach that limits the number of request each carrier can submit to no more than one. [Xu et al. \(2017\)](#) propose an effective auction mechanism for carriers to collaborate in bilateral lane exchange where carriers only submit the lane with the highest marginal cost. [Liu et al. \(2010b\)](#) introduce a non-auction request selecting and routing problem for making outsourcing request decisions and routing plans in collaborative transportation. [Hernández et al. \(2011\)](#) study the problem of seeking partners for single carrier to serve its excess demand. The impact of collaborative degree represented by discount rates on transportation cost is explored. Experiments reveal that collaboration of higher degree can increase the benefits.

Similarly, in the decentralized setting, existing research only considers how to exchange the requests efficiently. To the best of our knowledge, there is no research that studies the collaboration of heterogeneous depots, in which case the products owned by depots are different. (We use depots to represent companies that provide delivery services such as logistics providers, carriers, and express delivery companies.) However, in reality, collaboration of heterogeneous depots is common and often necessary. For instance, in the delivery network of JD, one of the e-commerce giants in China, there are multiple satellite depots in a metropolitan area that carry different products. In many cases, the demand of a customer can only be met by a specific depot. As a result, the satellite depots collaborate in product delivery and vehicle routing. To make the research more applicable to the realistic scenarios faced by the logistics providers in collaboration, we investigate the heterogeneous multi-depot collaborative vehicle routing problem (HMCVRP) in this paper. The aim of the research is to develop a mathematical programming model and an efficient algorithm for joint route planning and product transfer of multiple collaborative heterogeneous depots, and to shed light on the relation between the outcome of the collaboration and factors including locations of the depots and allocation of the collaboration benefits.

We use the transshipment mechanism to manage the physical transfer of products in collaboration of heterogeneous depots as order exchange is no longer sufficient ([Cortés et al., 2010](#)). Transshipment of products can take place at any depot and customer, which is called a transfer point (TP) in the logistics network. If a vehicle of depot A has customers of depot B in its route as the result of collaborative route planning, the vehicle may load depot B’s product at depot B or a customer of depot B’s. In the former case, depot B serves as a TP. In the latter, the customer is designated as a TP and depot B must send one of its vehicles to unload at this customer for the transfer to be possible.

The interplay of TP and route decisions drastically increases the modeling and computational complexity of HMCVRP. First, any node in the network could be selected as a TP. For every given TP designation, the route planning problem is a multi-depot VRP. Consequently, the solution space of HMCVRP is much larger than a classic VRP. Second, the arrival times of the vehicles that are engaged in product transshipment at the same customer node must be coordinated to make sure the loading vehicle arrives no earlier than the unloading vehicle. Hence, the vehicle routes are not completely independent to each other. Third, the product flow, in the form of each vehicle’s load, must be monitored throughout its route. A vehicle may receive new products as the result of

transshipments. It is thus necessary to ensure the vehicle's load does not exceed its capacity after each transshipment. The amount of product transshipped at each transfer point must also be carefully planned. All of above contribute to the unique challenges of HMCVRP.

In this paper, we formulate HMCVRP as a mixed-integer programming model with conditional constraints. The conditional constraints are used to check whether the arrival times and product flow requirements of the vehicles can be satisfied for potential TP and routing decisions. Taking advantage of the relation between routing constraints and the conditional constraints, we design a Benders-based branch-and-cut algorithm to solve HMCVRP. In particular, the technique of combinatorial Benders' cuts is incorporated into the algorithm to generate cuts from the conditional constraints. Enhancement strategies are developed to further speed up the algorithm. Computational study shows that the algorithm is significantly more efficient than the big-M transformation-based method. Results of additional experiments indicate the importance of the depots' location and a well-designed allocation mechanism of the cost savings.

The contributions of this paper are as follows. First, we formally define the heterogeneous multi-depot collaborative vehicle routing problem (HMCVRP) and formulate it as a mixed-integer program. In particular, products can be transshipped from one vehicle to another at the location of a depot or customer. The basic settings of HMCVRP and its MIP formulation can easily accommodate more complex settings and restrictions, and thus can serve as a modeling framework for various application-specific scenarios. Second, we develop a fast Benders-based branch-and-cut algorithm to solve HMCVRP. The algorithm incorporates the combinatorial Benders' cuts for solving mixed-integer programs with conditional constraints and shows superior computational efficiency over the big-M transformation methods. It exploits the unique yet generic structure of HMCVRP and has the potential to solve more complex variants of HMCVRP with minimal adjustments. Third, we provide some important operational insights of HMCVRP, including the relation between total cost savings and the distribution of the depots, and the impact of the allocation mechanism of the costs savings on individual depot's collaboration incentive. These insights can help the management at the enterprise level, if the depots are owned by one company, to better select, coordinate and incentivize the depots. The results are also valuable for the depots, if they operate independently, to better evaluate their individual benefits of collaboration.

The remainder of this paper is organized as follows. In Section 2, HMCVRP is defined and formulated as a mixed-integer programming problem with conditional constraints. In Section 3, we propose a Benders-based branch-and-cut algorithm with the combinatorial Benders cuts. Additional computational enhancement strategies are also developed. In Section 4, we test the algorithm and investigate the locations of depots and allocation of cost savings in HMCVRP by computational experiments. Section 5 concludes the paper.

2. HMCVRP

2.1. Problem description

Consider multiple depots (or logistics providers) in a given area that deliver products to their customers. Denote the set of the depots by O . Suppose these depots are heterogeneous: each depot provides a unique type of product. We use $e \in O$ to refer to a depot and the product it carries interchangeably without causing ambiguity, unless otherwise specified. Without loss of generality, assume each customer in the area only orders one type of product, and therefore, is only serviced by one depot. For a depot $e \in O$, denote the set of its customers by C_e , and let $N_e = \{e\} \cup C_e$. Let $C = \bigcup_{e \in O} C_e$ be the set of all customers and $N = \bigcup_{e \in O} N_e = C \cup O$ be the collection of all depots and customers. Note $C_e \cap C_{e'} = \emptyset$ and $N_e \cap N_{e'} = \emptyset$ for all $e, e' \in O, e \neq e'$. Consider the logistics network $G = (N, E)$ where N is the node set, and the edge set $E = \{(i, j) : i, j \in N\}$ represents the paths between the nodes. The demand of a customer $i \in C$ is d_i . Each depot e owns a fleet of vehicles, denoted by K_e . Let $K = \bigcup_{e \in O} K_e$ be the set of all vehicles. Use $\bar{K}_e = K \setminus K_e$ to denote the set of vehicles not belonging to depot e . For simplicity, assume each vehicle $k \in K$ has the same capacity Q , and assume the travel cost between nodes i and j is equal to the travel time c_{ij} . The goal of the problem is to find the optimal vehicle routes with the least possible total travel cost, while meeting every customer's demand.

If the depots plan their delivery routes individually, each will simply solve a vehicle routing problem (VRP). However, if the depots can collaborate to jointly generate their routes, their total travel costs can be reduced. Conceptually, if a customer $i \in C_{e'}$ is located near to some customers of another depot e , then it could be more cost effective to have a vehicle $k \in K_e$ take a short detour to service i . This might save an otherwise long trip for a vehicle from the depot e' by only adding a small travel cost to the route of vehicle k . In this paper, we formally propose this heterogeneous multi-depot collaborative vehicle routing problem (HMCVRP) where the depots collaboratively plan the routes of their vehicles.

The depots must collaborate in two dimensions: First, the product of one depot must be transferred to vehicles of another depot whenever necessary. This can be implemented via the method of transshipment. Specifically, transshipment refers to the practice that a vehicle $k \in K_e$ loads a different product e' at a pre-selected node called transfer point (TP). In this paper we assume such TP must be depot e' or one of its customers. Second, the routes of all vehicles should be carefully planned to coordinate the product transshipment and delivery. For instance, if the TP is a customer of depot e' , then a vehicle $k' \in K_{e'}$ must unload product e' before vehicle k arrives at the TP. This can be done by a centralized planner if all depots are owned by one company (in JD's case), or by a logistics platform that coordinates the depots (for example, the business model of Alibaba's Rookie Station).

Fig. 1 is an illustrative example of the vehicle routes as a result of the collaboration between two depots. Each depot owns one vehicle. Customers 1, 2, 3 and 4 order product A. Depot B's customers are 5, 6, 7 and 8. One can observe that customer 3 is closer to depot B's customers 5 and 6 than to depot A and its customers. Fig. 1a shows the delivery routes when depots A and B do not collaborate. Fig. 1b shows the routes with collaboration. Notice that in Fig. 1b, customer 3 is serviced by depot B's vehicle who

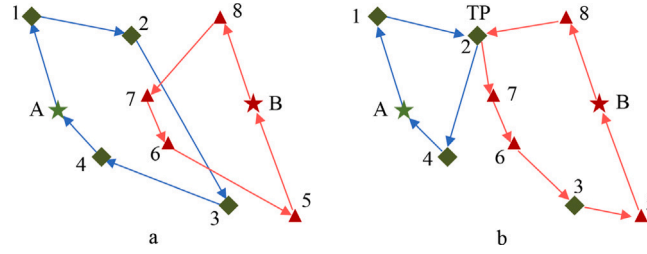


Fig. 1. Example of collaborative vehicle routing and transshipment.

loads product A at customer 2. In this case, depot B's vehicle adds node 2 and 3 to its route. The extra cost incurred is less than the savings of depot A's vehicle by not having to travel to 3. Consequently, the total travel cost of both depots is reduced.

In this research, we adopt the following general assumptions of the capacitated vehicle routing problem: Each depot has sufficient product and vehicles to satisfy its customers. Each vehicle starts and ends its route at the depot it belongs to. Each customer's order must be delivered by only one vehicle. A vehicle can be assigned to no more than one route. No customer has higher demand than the vehicle's capacity Q . Time spent on loading and unloading is negligible.

In addition, we make the following assumptions to simplify the complexity of HMCVRP without loss of its generality: First, if a customer node is designated as a TP, the vehicle that services the customer must belong to the depot that carries the product this customer orders. Moreover, only this vehicle is allowed to unload and only one vehicle is allowed to load at this node. This assumption implies that the transshipped product, if any, at a customer node must be of the same type as the customer's demand. This assumption also limits at most one transshipment event can take place at any customer node. Second, no vehicle is allowed to unload at any depot. In this case, if a depot is a TP, the transshipped product must be the depot's product. Unlike a customer TP, there is no restriction of the number of vehicles that can load at a depot TP. Third, a vehicle can load at no more than one TP. This assumption is imposed not only for reducing modeling and computational complexity, but because of two practical considerations. On one hand, by limiting the number of loading visits of each vehicle, we aim to avoid excessively long routes and to ensure timely delivery for all customers. On the other hand, to ensure customer satisfaction, we implicitly prioritize the service of each depot's own customers over loading other products and service of other depot's customers. There is no restriction of the number of TPs where a vehicle can unload. Although these assumptions reduce the complexity of the HMCVRP in this paper, our model and algorithm can be extended to a more general case in which the above assumptions are relaxed or removed.

2.2. HMCVRP formulation

We formulate the HMCVRP as a mixed-integer program (MIP) with conditional constraints. For any depot e , node $i \in N_e$ and vehicle $\bar{k} \in \bar{K}_e$, define a binary variable $z_i^{\bar{k}}$ that takes 1 if vehicle $\bar{k}$ loads at node i and 0 otherwise. Define a binary variable x_{ij}^k which equals 1 if vehicle k travels from node i to j in its route, and 0 otherwise. Let q_i be the transshipment quantity at node i . Define Q_{ie}^k as the quantity of depot e 's product carried by vehicle k when it leaves node i . Let t_i^k be vehicle k 's arrival time at node i . With these variables, HMCVRP can be formulated as the following problem (P1):

$$P1: \min \sum_{k \in K} \sum_{i, j \in N} c_{ij} x_{ij}^k \quad (1)$$

$$s.t. \sum_{k \in K} \sum_{j \in N} x_{ij}^k = 1 + \sum_{\bar{k} \in \bar{K}_e} z_i^{\bar{k}}, \quad \forall e \in O, i \in C_e \quad (2)$$

$$\sum_{k \in K_e} \sum_{j \in N} x_{ij}^k \leq 1, \quad \forall e \in O, i \in C_e \quad (3)$$

$$\sum_{j \in N} x_{ij}^k = \sum_{j \in N} x_{ji}^k, \quad \forall i \in N, k \in K \quad (4)$$

$$z_e^{\bar{k}} = \sum_{j \in N} x_{ej}^{\bar{k}}, \quad \forall e \in O, \bar{k} \in \bar{K}_e \quad (5)$$

$$z_i^{\bar{k}} \leq \sum_{j \in N} x_{ij}^{\bar{k}}, \quad \forall e \in O, i \in C_e, \bar{k} \in \bar{K}_e \quad (6)$$

$$z_i^{\bar{k}} \leq \sum_{k \in K_e} \sum_{j \in N} x_{ij}^k, \quad \forall e \in O, i \in C_e, \bar{k} \in \bar{K}_e \quad (7)$$

$$\sum_{\bar{k} \in \bar{K}_e} z_i^{\bar{k}} \leq 1, \quad \forall e \in O, i \in C_e \quad (8)$$

$$\sum_{j \in N} x_{ej}^k \leq 1, \quad \forall e \in O, k \in K_e \quad (9)$$

$$\sum_{i \in N \setminus N_e} z_i^k \leq 1, \quad \forall e \in O, \bar{k} \in K_e \quad (10)$$

$$x_{ij}^k = 1 \Rightarrow t_i^k + c_{ij} \leq t_j^k, \quad \forall e \in O, k \in K_e, i \in N, j \in N \setminus \{e\} \quad (11)$$

$$z_i^k = 1 \Rightarrow t_i^k \leq t_i^{\bar{k}}, \quad \forall e \in O, k \in K_e, \bar{k} \in \bar{K}_e, i \in C_e \quad (12)$$

$$x_{ij}^k = 1 \Rightarrow Q_{ie}^k - Q_{je}^k = 0, \quad \forall e \in O, k \in K, i \in N, j \in N \setminus N_e \quad (13)$$

$$x_{ij}^k = 1 \Rightarrow Q_{je}^k + d_j + q_j - Q_{ie}^k \leq 0, \quad \forall e \in O, k \in K_e, i \in N, j \in C_e \quad (14)$$

$$x_{ij}^{\bar{k}} = 1 \Rightarrow Q_{ie}^{\bar{k}} - (1 - z_j^{\bar{k}})d_j + q_j - Q_{je}^{\bar{k}} = 0, \quad \forall e \in O, \bar{k} \in \bar{K}_e, i \in N, j \in N_e \quad (15)$$

$$q_i \leq Q * \sum_{\bar{k} \in \bar{K}_e} z_i^{\bar{k}}, \quad \forall e \in O, i \in C_e \quad (16)$$

$$\sum_{\bar{e} \in O \setminus \{e\}} Q_{e\bar{e}}^k = 0, \quad \forall e \in O, k \in K_e \quad (17)$$

$$\sum_{e \in O} Q_{ie}^k \leq Q, \quad \forall k \in K, i \in N \quad (18)$$

$$x_{ij}^k \in \{0, 1\}, \quad \forall i, j \in N, k \in K \quad (19)$$

$$z_i^k \in \{0, 1\}, \quad \forall e \in O, i \in N_e, \bar{k} \in \bar{K}_e \quad (20)$$

$$t_i^k \geq 0, \quad \forall i \in N, k \in K \quad (21)$$

$$q_i, Q_{ie}^k \geq 0, \quad \forall e \in O, i \in N, k \in K \quad (22)$$

The objective function (1) is the total travel cost. Constraint (2) means that each customer is visited by the vehicle that delivers the customer's order and any vehicle that loads product at the node if it is a TP. Constraint (3) specifies that a depot sends no more than one vehicle to any of its customers. Constraint (4) is the flow balancing constraint. Constraint (5) means a depot must be a TP if it is visited by any other depot's vehicle, and vice versa. Constraint (6) specifies if a customer is visited by other depot's vehicle, the customer is either a TP or being serviced by the vehicle. Constraint (7) ensures that for a customer to be a TP, there must be a vehicle that unloads at the node. Constraint (8) specifies that at most one vehicle can load products at any customer node. Constraints (2) and (8) imply that each customer must be visited either once or twice, depending on whether the customer is a TP. Constraint (9) means that each vehicle can be used at most once. Constraint (10) requires that each vehicle can load at no more than one TP. Constraints (11)–(15) are called conditional constraints that only need to be satisfied when certain variables take specific values (1 in HMCVRP's case). Constraint (11) ensures the arrival times of a vehicle at the nodes in its route are consistent with the travel times. Constraint (12) enforces that the unloading vehicle arrives before the loading vehicle at any customer TP. Note that these two time windows constraints are used to regulate transshipment behaviors of the vehicles rather than to reflect the customers' service requirements. Constraints (13)–(15) keep track of the quantity of each product that a vehicle carries throughout its route. Constraint (16) allows the transshipment quantity at a customer node to be positive only if some vehicle comes to load. Constraint (17) means a vehicle does not carry any other type of product when it departs its depot. Constraint (18) prevents the vehicle capacity from being violated.

Table 1
Transformation of conditional constraints.

Constraint	Transformation	
(11)	$t_i^k + c_{ij} - t_j^k \leq (1 - x_{ij}^k) * M$	$\forall e \in O, i \in N, j \in N \setminus \{e\}, k \in K_e \quad (11a)$
(12)	$t_i^k - t_i^{\bar{k}} \leq (1 - z_i^{\bar{k}}) * M$	$\forall e \in O, i \in C_e, k \in K_e, \bar{k} \in \bar{K}_e \quad (12a)$
(13)	$(Q_{ie}^k - Q_{je}^k) \leq (1 - x_{ij}^k) * M$	$\forall e \in O, k \in K, i \in N, j \in N \setminus N_e \quad (13a)$
	$(Q_{je}^k - Q_{ie}^k) \leq (1 - x_{ij}^k) * M$	$\forall e \in O, k \in K, i \in N, j \in N \setminus N_e \quad (13b)$
(14)	$Q_{je}^k + d_j + q_j - Q_{ie}^k \leq (1 - x_{ij}^k) * M$	$\forall e \in O, k \in K_e, i \in N, j \in C_e \quad (14a)$
(15)	$Q_{ie}^k - (1 - z_j^{\bar{k}})d_j + q_j - Q_{je}^{\bar{k}} \leq (1 - x_{ij}^{\bar{k}}) * M$	$\forall e \in O, \bar{k} \in \bar{K}_e, i \in N, j \in N_e \quad (15a)$
	$Q_{je}^k + (1 - z_j^{\bar{k}})d_j - q_j - Q_{ie}^k \leq (1 - x_{ij}^{\bar{k}}) * M$	$\forall e \in O, \bar{k} \in \bar{K}_e, i \in N, j \in N_e \quad (15b)$

The formulation of problem P1 clearly shows that HMCVRP is significantly more complex and challenging than its homogeneous counterpart. To model the physical act of transshipment, we introduce z_i^k to indicate whether node i is selected as TP, t_i^k to track the vehicle arrival times, and q_i to denote the quantity of transferred product at node i . More constraints are needed in P1 to model the requirements of these new variables and the additional restrictions they impose on the routing variables x_{ij}^k . Consequently, solving P1 could be computationally extremely expensive. More specifically, the conditional constraints (11)–(15) must be properly treated. A theoretically feasible approach is to reformulate P1 into a mixed-integer linear program (MILP) that can be solved by CPLEX or any other MILP solver. Specifically, using a large number M , constraints (11)–(15) can be transformed to linear inequalities as shown in Table 1. The reformulated problem consists of the objective function (1), constraints (2)–(10), (16)–(22) and (11a)–(15b). However, as we will show in Section 4, this approach is too inefficient to be useful in practice. The disadvantage of having the big-M

Table 2

Features of CBC algorithms in recent literature and this paper.

Reference	Model decomposed	#Subproblems	MIS	Enhancement strategies
Bai and Rubin (2009)	Relaxed	1	N/A	Important customizations are introduced.
Chen et al. (2012)	Original	1	N/A	Constraints are added to the master problem.
Taşkın and Cevik (2013)	Original	1	Duality theory	Problem-specific valid inequalities and heuristics.
Côté et al. (2014)	Original	1	Greedy method	Speeding up techniques.
Verstichel et al. (2015)	Original	1	Enumeration method	N/A
Côté et al. (2021)	Relaxed	1	Heuristic approach.	Problem-specific methods.
Bruglieri et al. (2021)	Original	1	N/A	Multiple CBC approaches for stronger cuts.
Mancini et al. (2021a)	Relaxed	1	N/A	Pre-processing procedure.
This paper	Original	2	Duality theory	Subtour elimination; Single vehicle subproblem; additional VRP cuts.

constraints is that, intuitively speaking, they become less effective or ineffective in the relaxed problems throughout the branch-and-bound process, and thus cause excess computational burden. Moreover, the problem size grows rapidly when the numbers of depots, customers and vehicles increase. Benders' decomposition can be an ideal technique for such type of large-scale problems. Unfortunately, the standard Benders-based branch-and-cut method on the reformulation with transformed constraints of P1 would suffer the same computational disadvantage as the solvers because the generated Benders' cuts contain the big-M terms. In this research, we develop an algorithm that overcomes the above two challenges to solve HMCVRP much more efficiently.

3. Algorithm

We design an exact algorithm based on the Benders-based branch-and-cut framework (for example, McDaniel and Devine 1977, Naoum-Sawaya and Elhedhli 2013) to solve HMCVRP. Benders-based branch-and-cut (BBC) methods are broadly used to solve complex (mixed-)integer linear programming problems. (See Adulyasak et al. 2015, Çalık and Fortz 2019, Elçi and Noyan 2018 and Taherkhani et al. 2020 for applications.) In our algorithm, P1 is decomposed into a master problem (an integer program) and two subproblems (both linear programs) by the Benders decomposition methodology (Benders, 1962). The master problem finds optimal routes and TP locations without considering the vehicles' arrival times at the nodes or tracking the product quantities carried by each vehicle. By including the active conditional constraints, the subproblems check whether the optimal solution to the master problem is feasible for P1. If either of the subproblems is infeasible, new cuts will be added to the master problem. While these cuts could be generated in various ways, their form and quality can largely dictate the efficiency of the algorithm. In this research, we employ the technique of “combinatorial Benders' cuts” that has shown superior computational performance for MILPs with conditional constraints (Codato and Fischetti, 2006). We also develop a series of problem-specific enhancement strategies to further speed up the algorithm.

The combinatorial Benders' cuts (CBC) based algorithms have been successfully applied to solve various problems, with problem-specific designs and enhancement strategies. We review the features of the CBC algorithms used in some recent literature in Table 2, including the model being decomposed (original vs. relaxed), number of subproblems, whether the minimal infeasible subsystem (MIS) is identified, and the enhancement strategies used. We also compare them with the algorithm in this paper. The unique innovation of the CBC algorithm in this research is enabled by utilizing some state-of-the-art methods in VRP research which has successfully improved the performance of our algorithm. More specifically, two subproblems SP1 and SP2 are formulated to check whether the requirements of arriving time and vehicles capacity are satisfied, respectively, which can generate more cuts than using one subproblem. Furthermore, a new subproblem (SP3) is formulated for each vehicle as an enhancement strategy for quickly generating more cuts. To generate high-quality, tighter cuts, we use the duality theory to identify the MIS. Additionally, the subtour elimination algorithm is integrated as another enhancement strategy for speeding up the algorithm.

3.1. Benders-based branch-and-cut framework

Besides routing, the constraints of P1 address two more major considerations in collaborative route planning: arrival times and product flows. Arrival times of two vehicles that are engaged in transshipment at a customer TP must be coordinated. Product flow in the network is realized as the amount of each product that a vehicle carries and how it changes as the vehicle travels. To tackle the complexity of these three coupling groups of constraints, we decompose P1 into three parts: a master problem (MASTER) and two subproblems (SP1 and SP2).

$$\begin{aligned}
 \text{MASTER : } & \min \sum_{k \in K} \sum_{i, j \in N} c_{ij} x_{ij}^k \\
 & \text{s.t. Constraints (2) } \sim \text{(10), (19), (20).} \\
 \text{SP1 : } & \min 0 \\
 & \text{s.t. Constraints (11), (12), (21),} \\
 & \quad z_i^{\bar{k}} = 1 \Rightarrow t_i^{\bar{k}} \leq t_j^{\bar{k}} \quad \forall e \in O, \bar{k} \in \overline{K_e}, i \in N_e, j \in C_e \\
 \text{SP2 : } & \min 0 \\
 & \text{s.t. Constraints (13) } \sim \text{(18), (22).}
 \end{aligned} \tag{23}$$

The master problem aims to minimize the total travel cost of the routes with properly selected TPs under the assumption that its optimal solution, denoted by $(\bar{x}, \bar{z})$, meet the arrival times and product flow requirements. That is, the unloading vehicle must arrive before a loading vehicle at any customer TP, and every vehicle has the necessary amount of product on board for delivery or unloading purposes. We check these feasibility requirements by using the subproblems. SP1 attempts to coordinate the arrival times at each node for all vehicles. Constraint (23) is included to explicitly enforce that a vehicle cannot deliver to any customer of another depot's before loading the product, which was implied by constraints (14) and (15) in problem P1. SP2 seeks to arrange the quantity of every product that each vehicle carries at any point in its route. If SP1 and SP2 can be feasible for the MASTER optimal solution $(\bar{x}, \bar{z})$, then $(\bar{x}, \bar{z})$ is also feasible, and therefore optimal, for the original problem (P1). Note that the objective functions of SP1 and SP2 can be arbitrary (we use the constant zero) because the subproblems only address the feasibility of the MASTER solution and do not need to check its optimality. If either SP1 or SP2 is infeasible, then new cuts must be generated and included in MASTER as additional restrictions on its solution. The method for deriving such cuts, formally called combinatorial Benders' cuts, is explained in the next section.

We incorporate this Benders decomposition approach into a branch-and-bound framework. Without causing ambiguity, at every node of the branching tree, we still use MASTER to denote the master problem with the branching conditions included as additional constraints. At a node of the branching tree, we solve the LP relaxation of MASTER to obtain the node's lower bound. If an integer optimal solution is also obtained, we solve SP1 and SP2 at the same time to determine whether new cuts need to be generated. If so, we add new combinatorial Benders' cuts to MASTER. Note that the same cuts generated in SP1 and SP2 are added only once. Otherwise, the integer optimal solution is also optimal for this node, and it provides P1 with an upper bound. Note that the combinatorial Benders' cuts are global cuts. That is, any combinatorial Benders' cut, after being generated, will be added to the MASTER at every node of the remainder of the branching tree. To speed up the algorithm, we develop enhancement strategies (details in Section 3.3.) to generate additional cuts for the MASTER to recover the restrictions on the routing and TP variables that are implicitly enforced by constraints (11)–(18).

Branching strategy. At a node of the branching tree, if the optimal solution to the relaxed master problem is not integer, then for every depot e we calculate its “TP number” $\sum_{i \in N_e} \sum_{k \in \bar{K}_e} z_i^k$ which denotes the number of associated nodes (customers and depot itself) that are designated as TPs. Branch on the TP number whose fractional part is closest to 0.5. If the TP number of every depot is an integer, we then calculate each depot's “vehicle number” $\sum_{k \in K_e} \sum_{j \in N} x_{ej}^k$ which denotes the number of vehicles used by the depot. Branch on the vehicle number with fractional part closest to 0.5. If every vehicle number is integer, we then branch on variable $\bar{z}_i^k$. If every z_i^k is an integer, we branch on variable x_{ij}^k .

3.2. Combinatorial Benders' cuts

The technique of combinatorial Benders' cuts is proposed by Codato and Fischetti (2006) for efficiently solving mixed-integer programs with conditional constraints (Chen et al., 2018). It is shown to be more computationally efficient than big-M transformation-based methods. Conceptually, combinatorial Benders' cuts are generated by solving a subproblem (or multiple subproblems) and are added to the master problem (a mixed integer program) to explicitly remove the infeasible combinations of the integer variables. If incorporated into a branch-and-cut scheme, the combinatorial Benders' cuts can lead to tighter LP relaxation of the original integer problem with conditional constraints than its big-M transformed reformulation. Moreover, the size of the branching tree can be significantly reduced because the combinatorial Benders' cuts can prevent some infeasible nodes from being created. In this section, we describe how combinatorial Benders' cuts are constructed in the proposed algorithm for HMCVRP.

At every node of the branching tree, if the optimal solution to the linear relaxation of the master problem is integer, we examine the dual problems of SP1 and SP2. Formally, the dual problem (DP1) of SP1 is as follows.

$$\max - \sum_{e \in O} \sum_{i \in N} \sum_{j \in N \setminus \{e\}} \sum_{k \in K_e} \bar{x}_{ij}^k c_{ij} \alpha_{kij}^e \quad (24)$$

$$\text{s.t.} \quad \sum_{j \in N \setminus \{e\}} \bar{x}_{ej}^k \alpha_{kej}^e \leq 0, \quad \forall e \in O, k \in K_e \quad (25)$$

$$\sum_{j \in N \setminus \{e\}} \bar{x}_{ij}^k \alpha_{kij}^e - \sum_{j \in N} \bar{x}_{ji}^k \alpha_{kji}^e + \sum_{\bar{k} \in \bar{K}_e} \bar{z}_i^{\bar{k}} \beta_{k\bar{k}}^e \leq 0, \quad \forall e \in O, k \in K_e, i \in C_e \quad (26)$$

$$\sum_{j \in N \setminus \{\bar{e}\}} \bar{x}_{ej}^k \alpha_{kej}^{\bar{e}} - \sum_{j \in N} \bar{x}_{je}^k \alpha_{kje}^{\bar{e}} + \sum_{j \in C_e} \bar{z}_e^j \gamma_{kej}^e \leq 0, \quad \forall e \in O, \bar{e} \in O \setminus \{e\}, k \in K_{\bar{e}} \quad (27)$$

$$\sum_{j \in N \setminus \{\bar{e}\}} \bar{x}_{ij}^k \alpha_{kij}^{\bar{e}} - \sum_{j \in N} \bar{x}_{ji}^k \alpha_{kji}^{\bar{e}} - \sum_{k \in K_e} \bar{z}_i^k \beta_{k\bar{k}}^e + \sum_{j \in C_e} \bar{z}_i^j \gamma_{kji}^e - \sum_{j \in N_e} \bar{z}_j^k \gamma_{kji}^e \leq 0, \quad \forall e \in O, i \in C_e, \bar{e} \in O \setminus \{e\}, \bar{k} \in K_{\bar{e}} \quad (28)$$

$$\alpha_{kij}^e \leq 0, \quad \forall e \in O, k \in K_e, i \in N, j \in N \setminus \{e\} \quad (29)$$

$$\beta_{k\bar{k}}^e \leq 0, \quad \forall e \in O, k \in K_e, \bar{k} \in \bar{K}_e, i \in C_e \quad (30)$$

$$\gamma_{kij}^e \leq 0, \quad \forall e \in O, \bar{k} \in \bar{K}_e, i \in N_e, j \in C_e \quad (31)$$

where $(\bar{x}, \bar{z})$ is the optimal solution to the master problem (MASTER) and α, β, γ are vectors of dual variables associated with constraints (11), (12) and (23), respectively. Note that when $\bar{x}_{ij}^k = 0$, the corresponding α_{kij}^e plays no effective role in DP1 and may

take any numerical value. Similarly, when $\bar{z}_i^k = 0$, the corresponding β_{kij}^e and γ_{kij}^e may take any value without affecting the solution to DP1.

Note that DP1 is always feasible because $(\alpha, \beta, \gamma) = (0, 0, 0)$ is a feasible solution. If DP1 is bounded, SP1 must be feasible. If DP1 is unbounded, then there exists a feasible solution $(\alpha^*, \beta^*, \gamma^*)$ such that $-\sum_{e \in O} \sum_{i \in N} \sum_{j \in N \setminus \{e\}} \sum_{k \in K_e} \bar{x}_{ij}^k c_{ij}^e \alpha_{kij}^{e*} > 0$. In this case, $(\bar{x}, \bar{z})$ will not be feasible for the original problem P1. By the combinatorial Benders' cuts technique, the following cut needs to be added to the master problem:

$$\sum_{(i,j,k) \in J_{\bar{x}}} (1 - x_{ij}^k) + \sum_{(i,k) \in J_{\bar{z}}} (1 - z_i^k) \geq 1 \quad (32)$$

where the index set $J_{\bar{x}}$ and $J_{\bar{z}}$, formally defined as below, refer to those variables x_{ij}^k and z_i^k with at least one associated dual variable being not zero. The cut (32) means that at least one variable in $(\bar{x}, \bar{z})$ must switch from 1 to 0.

$$J_{\bar{x}} = \left\{ (i, j, k) \mid \bar{x}_{ij}^k = 1, i \in N \text{ and there exists } e \in O \text{ such that } j \in N \setminus \{e\}, k \in K_e \text{ and } \alpha_{kij}^{e*} < 0 \right\}.$$

$$J_{\bar{z}} = \left\{ (i, \bar{k}) \mid \bar{z}_i^{\bar{k}} = 1, \text{ and there exists } e \in O \text{ such that } i \in C_e, \bar{k} \in \bar{K}_e \text{ and } \beta_{kij}^{e*} \neq 0 \text{ for some } k \in K_e \text{ or } i \in N_e, \bar{k} \in \bar{K}_e \text{ and } \gamma_{kij}^{e*} \neq 0 \text{ for some } j \in C_e \right\}.$$

In a similar manner, we derive the combinatorial Benders' cut that is generated by SP2. The dual problem DP2 of SP2 is as below.

$$\max - \sum_{e \in O} \sum_{k \in K_e} \sum_{i \in N} \sum_{j \in C_e} \bar{x}_{ij}^k d_j \psi_{kij}^e + \sum_{e \in O} \sum_{k \in \bar{K}_e} \sum_{i \in N} \sum_{j \in N_e} \bar{x}_{ij}^k (1 - \bar{z}_j^{\bar{k}}) d_j \rho_{kij}^e + \sum_{e \in O} \sum_{i \in C_e} \sum_{k \in \bar{K}_e} Q \bar{z}_i^{\bar{k}} \tau_i + \sum_{k \in K} \sum_{i \in N} Q \omega_{ki} \quad (33)$$

$$\text{s.t. } \sum_{\bar{k} \in \bar{K}_e} \sum_{i \in N} \bar{x}_{ie}^{\bar{k}} \rho_{kie}^e \leq 0, \quad \forall e \in O \quad (34)$$

$$\sum_{k \in K_e} \sum_{i \in N} \bar{x}_{ij}^k \psi_{kij}^e + \sum_{\bar{k} \in \bar{K}_e} \sum_{i \in N} \bar{x}_{ij}^{\bar{k}} \rho_{kij}^e + \tau_j \leq 0, \quad \forall e \in O, j \in C_e \quad (35)$$

$$\sum_{j \in N \setminus N_e} \bar{x}_{ej}^k \zeta_{kej}^e - \sum_{j \in C_e} \bar{x}_{ej}^k \psi_{kej}^e + \omega_{ke} \leq 0, \quad \forall e \in O, k \in K_e \quad (36)$$

$$\sum_{j \in N \setminus N_e} \bar{x}_{ij}^k \zeta_{kij}^e - \sum_{j \in C_e} \bar{x}_{ij}^k \psi_{kij}^e + \sum_{j \in N_e} \bar{x}_{ji}^k \psi_{kji}^e + \omega_{ki} \leq 0, \quad \forall e \in O, k \in K_e, i \in C_e \quad (37)$$

$$\sum_{j \in N \setminus N_e} \bar{x}_{ij}^k \zeta_{kij}^e - \sum_{j \in N_e} \bar{x}_{ji}^k \zeta_{kji}^e - \sum_{j \in C_e} \bar{x}_{ij}^k \psi_{kij}^e + \omega_{ki} \leq 0, \quad \forall e \in O, k \in K_e, i \in N \setminus N_e \quad (38)$$

$$\sum_{j \in N \setminus N_e} \bar{x}_{ij}^k \zeta_{kij}^e + \sum_{j \in N_e} \bar{x}_{ji}^k \rho_{kij}^e - \sum_{j \in N} \bar{x}_{ji}^k \rho_{kji}^e + \omega_{\bar{k}i} \leq 0, \quad \forall e \in O, \bar{k} \in \bar{K}_e, i \in N_e \quad (39)$$

$$\sum_{j \in N \setminus N_e} \bar{x}_{ij}^k \zeta_{kij}^e - \sum_{j \in N} \bar{x}_{ji}^k \zeta_{kji}^e + \sum_{j \in N_e} \bar{x}_{ij}^k \rho_{kij}^e + \omega_{\bar{k}i} \leq 0, \quad \forall e \in O, \bar{k} \in \bar{K}_e, i \in C \setminus C_e \quad (40)$$

$$\sum_{j \in N \setminus N_e} \bar{x}_{ij}^k \zeta_{kij}^e - \sum_{j \in N} \bar{x}_{ji}^k \zeta_{kji}^e + \sum_{j \in N_e} \bar{x}_{ij}^k \rho_{kij}^e + \omega_{\bar{k}i} \leq 0, \quad \forall e \in O, i \in O \setminus \{e\}, \bar{k} \in \bar{K}_e \setminus K_i \quad (41)$$

$$\sum_{j \in N \setminus N_e} \bar{x}_{ij}^k \zeta_{kij}^e - \sum_{j \in N} \bar{x}_{ji}^k \zeta_{kji}^e + \sum_{j \in N_e} \bar{x}_{ij}^k \rho_{kij}^e + \varphi_{\bar{k}} + \omega_{\bar{k}i} \leq 0, \quad \forall e \in O, i \in O \setminus \{e\}, \bar{k} \in K_i \quad (42)$$

$$\psi_{kij}^e \leq 0, \quad \forall e \in O, k \in K_e, i \in N, j \in C_e \quad (43)$$

$$\tau_i \leq 0, \quad \forall i \in C \quad (44)$$

$$\omega_{ki} \leq 0, \quad \forall k \in K, i \in N \quad (45)$$

where $(\bar{x}, \bar{z})$ is the optimal solution to the master problem (MASTER) and $\zeta, \psi, \rho, \tau, \varphi, \omega$ are vectors of dual variables associated with constraints (13)–(18), respectively. DP2 is always feasible since the dual variables all taking zeros form a feasible solution. Therefore, SP2 is feasible when DP2 is bounded, and SP2 is infeasible when DP2 is unbounded with a feasible solution $(\zeta^*, \psi^*, \rho^*, \tau^*, \varphi^*, \omega^*)$ making the objective function (33) positive. In this case, $(\bar{x}, \bar{z})$ is infeasible for P1, and the following cut will be added to the master problem:

$$\sum_{(i,j,k) \in I_{\bar{x}}} (1 - x_{ij}^k) + \sum_{(i,k) \in I_{\bar{z}}^1} (1 - z_i^k) + \sum_{(i,k) \in I_{\bar{z}}^2} z_i^k \geq 1 \quad (46)$$

where the index sets $I_{\bar{x}}$, $I_{\bar{z}}^1$ and $I_{\bar{z}}^2$ are defined as below.

$$I_{\bar{x}} = \left\{ (i, j, k) \mid \bar{x}_{ij}^k = 1, i \in N, \text{ and there exists } e \in O \text{ such that } j \in N \setminus N_e, \zeta_{kij}^{e*} \neq 0 \text{ for some } k \in K, \text{ or } j \in C_e, \psi_{kij}^{e*} < 0 \text{ for some } k \in K_e, \text{ or } j \in N_e, \rho_{kij}^{e*} \neq 0 \text{ for some } k \in \bar{K}_e \right\}.$$

$$I_{\bar{z}}^1 = \left\{ (i, \bar{k}) \mid \bar{z}_i^{\bar{k}} = 1, \text{ and there exists } j \in N, e \in O \text{ such that } i \in N_e, \bar{k} \in \bar{K}_e, \bar{x}_{ji}^{\bar{k}} = 1 \text{ and } \rho_{kji}^{e*} \neq 0 \right\}.$$

$$I_z^2 = \{ (i, \bar{k}) \mid \bar{z}_{ij}^k = 0, \text{ and there exists } j \in N, e \in O \text{ such that } i \in N_e, \bar{k} \in \bar{K}_e, \bar{x}_{ji}^{\bar{k}} = 1 \text{ and } \rho_{kji}^e \neq 0 \}.$$

3.3. Enhancement strategies

Generally speaking, if more valid cuts are added to the master problem, the lower bounds can be tighter, and the overall size of the branching tree can be smaller. Hence the branch-and-cut method can be computationally more efficient. We develop a series of enhancement strategies for this purpose.

3.3.1. Generate more cuts by DP1 and DP2

In DP1, each feasible $(\alpha^*, \beta^*, \gamma^*)$ with a positive value of (24) will lead to one combinatorial Benders' cut. To identify more of such solutions, we use the following variant of DP1, called DP3:

$$\max w_1^T \alpha + w_2^T \beta + w_3^T \gamma \quad (47)$$

$$\text{s.t.} - \sum_{e \in O} \sum_{i \in N} \sum_{j \in N \setminus \{e\}} \sum_{k \in K_e} \bar{x}_{ij}^k c_{ij} \alpha_{kij}^e = 1 \quad (48)$$

Constraints(25) ~ (31).

where w_1^T, w_2^T, w_3^T are coefficients that can be arbitrarily set. Any optimal solution to DP3 is a feasible solution to DP1 with a positive objective function value. Therefore, a new Benders' cut (32) may be generated when we use a different (w_1^T, w_2^T, w_3^T) in DP3. In the algorithm, for every $(\bar{x}, \bar{z})$ obtained from the master problem, DP3 is solved multiple times with different values in (w_1^T, w_2^T, w_3^T) . If different optimal solutions are recorded, as we have observed in numerical experiments, multiple combinatorial Benders' cuts can be generated.

Similarly, we formulate the following variant of DP2, called DP4:

$$\max w_4^T \xi + w_5^T \psi + w_6^T \rho + w_7^T \tau + w_8^T \phi + w_9^T \omega \quad (49)$$

$$\text{s.t.} - \sum_{e \in O} \sum_{k \in K_e} \sum_{i \in N} \sum_{j \in C_e} \bar{x}_{ij}^k d_j \psi_{kij}^e + \sum_{e \in O} \sum_{\bar{k} \in \bar{K}_e} \sum_{i \in N} \sum_{j \in N_e} \bar{x}_{ij}^{\bar{k}} (1 - \bar{z}_{ij}^{\bar{k}}) d_j \rho_{kij}^e + \sum_{e \in O} \sum_{i \in C_e} \sum_{\bar{k} \in \bar{K}_e} Q \bar{z}_i^{\bar{k}} \tau_i + \sum_{k \in K} \sum_{i \in N} Q \omega_{ki} = 1 \quad (50)$$

Constraints(34) ~ (45).

where the coefficients $w_4^T, \dots, w_9^T$ can be arbitrarily set. For every $(\bar{x}, \bar{z})$ of the master problem, we change the values of $w_4^T, \dots, w_9^T$ and solve DP4 multiple times to generate multiple combinatorial Benders' cuts.

3.3.2. Subtour elimination cuts

It is essential in any VRP formulation that subtours must be prohibited. This requirement is met in P1 by the constraints (11)~(18). More precisely, if the variables (x, z) satisfying constraints (2)~(10) contains subtours, there must exist a vehicle with a subtour that does not involve the depot it belongs to. It then would be impossible to arrange the arrival times or the product quantities, if not both, for the vehicle in this subtour. However, in the Benders decomposition method, the master problem (MASTER) does not involve constraints (11)~(18). There could be subtours in its optimal solution $(\bar{x}, \bar{z})$. Although the subtours will eventually be eliminated by the combinatorial Benders' cuts, the time it takes for the algorithm to completely generate these Benders' cuts could be unfavorably long. Therefore, we enhance the algorithm by adding a sub-routine that is specifically used to screen out subtours in $(\bar{x}, \bar{z})$ before passing it to SP1 and SP2.

A commonly used form of subtour elimination constraints in vehicle routing problems is as follows.

$$\sum_{i,j \in S} x_{ij}^k \leq \sum_{i \in S} \sum_{j \in N} x_{ij}^k - \sum_{j \in N} x_{sj}^k, \quad \forall e \in O, S \subseteq N \setminus \{e\}, k \in K_e, s \in S \quad (51)$$

The number of such constraints can easily explode when the size of the network and the number of vehicles grow. Hence, instead of adding all of the subtour elimination constraints to the master problem, we only add the ones that are violated by $(\bar{x}, \bar{z})$. We apply a min-cut algorithm-based method to detect subtours. For each vehicle, we extract the sub-graph covered by its route. Find the min-cut by the Stoer–Wagner algorithm (Stoer and Wagner, 1997). If the min-cut is less than 2, there must exist a subtour in the route. The violated subtour elimination cut is then identified and added to the master problem. We note that this procedure is not only applicable to integer $(\bar{x}, \bar{z})$, but also when $(\bar{x}, \bar{z})$ is not an integer solution. After all of the violated subtour elimination cuts are detected and added, we solve the master problem again to update $(\bar{x}, \bar{z})$. This process repeats until no subtour is detected in $(\bar{x}, \bar{z})$. By incorporating this enhancement strategy, we can significantly improve the lower bounds of the nodes of the branching tree. As a result, the branching tree can become smaller, and the algorithm converges faster.

3.3.3. Single vehicle subproblem

The only constraint in SP2 that involves multiple vehicles is constraint (16). The rest of the constraints are defined for each individual vehicle. To generate more Benders' cuts by utilizing this structure of SP2, we develop a single vehicle subproblem (SP3) for each vehicle. For vehicle k , SP3 is defined as follows.

$$\min 0 \quad (52)$$

$$s.t. \ x_{ij}^k = 1 \Rightarrow Q_{i\bar{e}}^k - Q_{j\bar{e}}^k = 0, \quad \forall \bar{e} \in O, i \in N, j \in N \setminus N_{\bar{e}} \quad (53)$$

$$x_{ij}^k = 1 \Rightarrow Q_{je}^k + d_j + q_j - Q_{ie}^k \leq 0, \quad \forall i \in N, j \in C_e \quad (54)$$

$$x_{ij}^k = 1 \Rightarrow Q_{i\bar{e}}^k - (1 - z_j^k)d_j + q_j - Q_{j\bar{e}}^k = 0, \quad \forall i \in N, \bar{e} \in O \setminus \{e\}, j \in N_{\bar{e}} \quad (55)$$

$$q_i \leq Q * z_i^k, \quad \forall i \in C \setminus C_e \quad (56)$$

$$\sum_{\bar{e} \in O \setminus \{e\}} Q_{e\bar{e}}^k = 0 \quad (57)$$

$$\sum_{\bar{e} \in O} Q_{i\bar{e}}^k \leq Q, \quad \forall i \in N \quad (58)$$

$$q_i, Q_{i\bar{e}}^k \geq 0, \quad \forall \bar{e} \in O, i \in N \quad (59)$$

Note constraint (56) relaxes constraint (16) by only involving z_i^k and dropping other vehicles. Therefore, the collection of feasible solutions to all single vehicle problems is not necessarily feasible for SP2. SP2 cannot be replaced by SP3.

The dual problem (DP5) of SP3 is below.

$$\max - \sum_{i \in N} \sum_{j \in C_e} \bar{x}_{ij}^k d_j \delta_{ij} + \sum_{i \in N} \sum_{j \in N \setminus N_e} \bar{x}_{ij}^k (1 - z_j^k) d_j \xi_{ij} + Q \sum_{i \in C \setminus C_e} \bar{z}_i^k \lambda_i + Q \sum_{i \in N} \eta_i \quad (60)$$

$$s.t. \ \sum_{i \in N} \bar{x}_{i\bar{e}}^k \xi_{i\bar{e}} \leq 0, \quad \forall \bar{e} \in O \setminus \{e\} \quad (61)$$

$$\sum_{i \in N} \bar{x}_{ij}^k \delta_{ij} \leq 0, \quad \forall j \in C_e \quad (62)$$

$$\sum_{i \in N} \bar{x}_{ij}^k \xi_{ij} + \lambda_j \leq 0, \quad \forall j \in C \setminus C_e \quad (63)$$

$$\sum_{j \in N \setminus N_e} \bar{x}_{ej}^k \theta_{ej}^e - \sum_{j \in C_e} \bar{x}_{ej}^k \delta_{ej} + \eta_e \leq 0 \quad (64)$$

$$\sum_{j \in N \setminus N_e} \bar{x}_{ej}^k \theta_{ej}^{\bar{e}} - \sum_{j \in N} \bar{x}_{je}^k \theta_{je}^{\bar{e}} + \sum_{j \in N_e} \bar{x}_{ej}^k \xi_{ej} + \mu + \eta_e \leq 0 \quad \forall \bar{e} \in O \setminus \{e\} \quad (65)$$

$$\sum_{j \in N \setminus N_e} \bar{x}_{ij}^k \theta_{ij}^e - \sum_{j \in C_e} \bar{x}_{ij}^k \delta_{ij} + \sum_{j \in N} \bar{x}_{ji}^k \delta_{ji} + \eta_i \leq 0, \quad \forall i \in C_e \quad (66)$$

$$\sum_{j \in N \setminus N_e} \bar{x}_{ij}^k \theta_{ij}^{\bar{e}} - \sum_{j \in N} \bar{x}_{ji}^k \theta_{ji}^{\bar{e}} + \sum_{j \in N_e} \bar{x}_{ij}^k \xi_{ij} + \eta_i \leq 0, \quad \forall \bar{e} \in O \setminus \{e\}, i \in N \setminus (\{e\} \cup N_{\bar{e}}) \quad (67)$$

$$\sum_{j \in N \setminus N_e} \bar{x}_{ij}^k \theta_{ij}^e - \sum_{j \in N} \bar{x}_{ji}^k \theta_{ji}^e + \sum_{j \in C_e} \bar{x}_{ij}^k \delta_{ij} + \eta_i \leq 0, \quad \forall i \in N \setminus N_e \quad (68)$$

$$\sum_{j \in N \setminus N_e} \bar{x}_{ij}^k \theta_{ij}^{\bar{e}} + \sum_{j \in N_e} \bar{x}_{ij}^k \xi_{ij} - \sum_{j \in N} \bar{x}_{ji}^k \xi_{ji} + \eta_i \leq 0, \quad \forall \bar{e} \in O \setminus \{e\}, i \in N_{\bar{e}} \quad (69)$$

$$\delta_{ij} \leq 0, \quad \forall i \in N, j \in C_e \quad (70)$$

$$\lambda_i \leq 0, \quad \forall i \in C \setminus C_e \quad (71)$$

$$\eta_i \leq 0, \quad \forall i \in N \quad (72)$$

Where $\theta, \delta, \xi, \lambda, \mu, \eta$ are vectors of dual variables associated with constraints (53)~(58), respectively. For an optimal solution $(\bar{x}, \bar{z})$ to the master problem, if DP5 is unbounded, then a combinatorial Benders' cut of the form of (46) for vehicle k will be added to the master problem. Since SP3 and DP5 are defined for every vehicle, the algorithm can generate more cuts than only using SP2 and DP2. Moreover, the technique in Section 3.3.1. can also be applied to DP5 to generate more cuts.

3.3.4. Additional cuts

The next two cuts are added to the master problem to remove undesired solutions. Constraint (73) below specifies that a vehicle $\bar{k}$ must visit a TP if it services other depots' customers. Constraint (74) ensure that a vehicle can visit a customer only if the vehicle leaves its depot. This constraint originates from Archetti et al. (2007) for studying inventory routing problems.

$$\sum_{j \in N} x_{ij}^k \leq \sum_{j \in N_e} z_j^{\bar{k}}, \quad \forall e \in O, i \in C_e, \bar{k} \in \bar{K}_e \quad (73)$$

$$\sum_{j \in N} x_{ij}^k \leq \sum_{j \in N} x_{ej}^k, \quad \forall e \in O, i \in C, k \in K_e \quad (74)$$

The next two cuts are used to break the symmetry of the routes. For vehicles belonging to the same depot, constraint (75) prevents vehicle $k+1$ from being used unless vehicle k is assigned any route. Constraint (76) use lexicographic ordering to uniquely number a set of customers in one route and ensures each route is assigned to a special vehicle. Adulyasak et al. (2014) show that

Table 3
Information of instances based on “r101”.

Solomon's benchmark	Group	#Depots	#Customers	Customer-depot assignment
r101	1	2	20	Instance 1~10: Distance; Instance 11~20: Random.
	2	2	30	
	3	3	20	
	4	3	30	

these symmetry-breaking constraints can significantly improve the computational performance.

$$\sum_{j \in N} x_{ej}^{k+1} \leq \sum_{j \in N} x_{ej}^k, \quad \forall e \in O, k \in K_e \setminus |K_e| \quad (75)$$

$$\sum_{i=|O|+1}^j 2^{j-i} \sum_{l \in N} x_{il}^{k+1} \leq \sum_{i=|O|+1}^j 2^{j-i} \sum_{l \in N} x_{il}^k, \quad \forall e \in O, j \in C, k \in K_e \setminus |K_e| \quad (76)$$

4. Computational study

We test the efficacy and computational efficiency of the proposed algorithm by numerical experiments. In particular, we compare its performance with solving the MILP reformulation of HMCVRP by CPLEX and investigate the benefits of each class of cuts. With exact solution to HMCVRP available, we then study the effect of the locations of the depots on the benefit of the collaborative vehicle routing. We also investigate the allocation of the total cost savings to each individual depot and whether a variety of such mechanisms can incentivize the depots to participate in the collaborative routing.

To the best of our knowledge, there are no existing HMCVRP benchmark problems. Hence, we generate HMCVRP instances based on the Solomon's VRPTW benchmark problems (Solomon, 1987). Specifically, we use problems “r101”, “c101” and “rc101”, each of which contains the location (x–y coordinates), demand and time window of 100 customers and the location of one depot. The time windows are discarded in our experiments because they are not considered in the HMCVRP. Based on each Solomon problem, we generate four groups of instances, with 20 instances in each group, by setting different numbers of depots and customers.

We describe the instances based on “r101”. In group 1, there are 2 depots (A and B) and 20 customers. The locations of depots A and B are (20, 35) and (50, 30), respectively. We take the customers 1 ~ 20 in the r101 Solomon problem and assign them to the depots in two ways, one named “Distance” and the other “Random”. In instances 1 ~ 10 where “Distance” method is applied, customer i is assigned to depot A with probability $p_{iA} = d_{iB}/(d_{iA} + d_{iB})$ where d_{iA} and d_{iB} are the Euclidean distances between i and depots A and B, respectively. That is, a customer is more likely to be assigned to the depot it is closer to. Similarly, customer i is assigned to depot B with probability $p_{iB} = d_{iA}/(d_{iA} + d_{iB})$. In instances 11 ~ 20 where “Random” method is applied, customer i is assigned to depots A and B with equal probability 0.5, i.e., completely randomly. In group 2, we use customers 1 ~ 30 in the r101 Solomon problem. The remaining settings are the same as in group 1. In group 3, there are 3 depots (A, B and C) and 20 customers (customer 1 ~ 20 of r101). The locations of depots A, B and C are (20, 22), (50, 20) and (35, 50), respectively. In instances 1 ~ 10, customer i is assigned to depots A, B and C with probabilities $p_{iA} = (d_{iB} + d_{iC})/(2 * (d_{iA} + d_{iB} + d_{iC}))$, $p_{iB} = (d_{iA} + d_{iC})/(2 * (d_{iA} + d_{iB} + d_{iC}))$ and $p_{iC} = (d_{iA} + d_{iB})/(2 * (d_{iA} + d_{iB} + d_{iC}))$, while in instances 11 ~ 20, the probabilities are all 1/3. In group 4, we use customer 1 ~ 30 of r101 and keep other settings the same as in group 3. Set the capacity of all vehicles in each instance to be 300. The information of these instances based on “r101” is summarized in Table 3. Description of instances based on Solomon problems “c101” and “rc101” are in the Appendix.

4.1. Efficacy of the algorithm

We solve each HMCVRP instance in groups 1 ~ 4 by five variants of the proposed Benders-based branch-and-cut algorithm: (1) the basic Benders-based branch-and-cut algorithm without any enhancement strategies, called “BBC”; (2) BBC with the “additional cuts” (Section 3.3.4) added in the master problem, called “BBC+”; (3) “BBC+SEC” which incorporates the subtour elimination strategy (Section 3.3.2) into BBC+; (4) “BBC+SVC” which incorporates the single vehicle subproblem strategy (Section 3.3.3) into BBC+; (5) “EBBC” which is fully enhanced BBC with all strategies in Section 3. We also solve the instances by calling the CPLEX solver for the big-M reformulation of HMCVRP (Section 2). This method is named “SVR” in this paper. These methods are implemented in Java with IBM ILOG CPLEX 12.10. All experiments are conducted in an Intel Core i7-7700 3.60 GHZ desktop with 16 GB memory. The maximum running time of each instance is set as 2 h. Note that the five variants of the proposed Benders-based branch-and-cut algorithm are exact algorithms, which will output the optimal solution if stopped within 2 h.

Tables 4, 5, 6 and 7 summarize the computational results of groups 1, 2, 3 and 4, respectively. To compare the computational performance of the above methods, we use the following metrics, all being the average of the 20 instances in each group: The lower bound (V) and its computing time (T) at the root node of the branch-and-bound tree are shown in row “RLB”. The number of cuts (N) generated by solving SP1, solving SP2, the subtour elimination strategy and the single vehicle subproblem, with their corresponding computing time (T), are reported in rows SP1C, SP2C, SEC and SVC, respectively. The total number of cuts (N) and computing time (T) are shown in row “All-C”. The total number of nodes processed in the branch-and-bound tree (row “Nodes”) and the computing time for solving the instance (row “CT”) are also recorded. All reported computing times are in seconds. Finally,

Table 4
Performance of different algorithms on instance group 1.

Group 1		BBC	BBC+	BBC+SEC	BBC+SVC	EBBC	SVR
RLB	V	238.82	241.32	247.01	242.11	247.73	n/a
	T(s)	0.16	0.23	0.3	0.26	0.36	
SP1C	N	3226.5	337.8	178.85	353.65	178.1	n/a
	T(s)	19.74	0.65	0.44	0.68	0.43	
SP2C	N	3429.2	249.65	862.1	0	0	n/a
	T(s)	15.66	0.87	0.55	0.05	0.03	
SEC	N	n/a	n/a	1146.6	n/a	1141.7	n/a
	T(s)	n/a	n/a	2.34	n/a	2.38	
SVC	N	n/a	n/a	n/a	260.25	145	n/a
	T(s)	n/a	n/a	n/a	0.82	0.53	
All-C	N	6655.7	587.45	2187.55	613.9	1464.8	n/a
	T(s)	35.4	1.52	3.33	1.55	3.37	
Nodes		102440.05	7916.6	1988.75	7556.25	2025.8	260147.2
CT(s)		518.03	12.86	10.83	12.65	10.91	300.11
#OPT		20	20	20	20	20	20
#FS		20	20	20	20	20	20

Table 5
Performance of different algorithms on instance group 2.

Group 2		BBC	BBC+	BBC+SEC	BBC+SVC	EBBC	SVR
RLB	V	315.85	317.01	322.88	317.03	323.61	n/a
	T(s)	0.34	0.42	0.56	0.46	0.67	
SP1C	N	12717.15	5104.95	2534.65	5119.1	2377.45	n/a
	T(s)	362.13	84.67	77.19	82.67	66.13	
SP2C	N	19451.45	7733.7	5270.8	0.95	0.95	n/a
	T(s)	231.03	46.31	44.02	0.1	0.1	
SEC	N	n/a	n/a	7918.6	n/a	8005.35	n/a
	T(s)	n/a	n/a	842.84	n/a	793.75	
SVC	N	n/a	n/a	n/a	7247.9	4578	n/a
	T(s)	n/a	n/a	n/a	42.33	31.63	
All-C	N	32168.6	12838.65	15724.05	12367.95	14961.75	n/a
	T(s)	593.16	130.99	964.05	125.1	891.6	
Nodes		142998.55	188831.2	57626.05	181162.85	48656.1	1443474.95
CT(s)		7200	4270.72	4471.84	3935.13	4135.43	6910.96
#OPT		0	11	13	12	12	1
#FS		19	20	20	20	20	18

we count the number of instances that are solved to optimality (row “#OPT”) and the number of those for which a feasible solution is obtained (row “#FS”) in each group.

Comparing EBBC and SVR, it is clear that the proposed algorithm with the enhancement strategies significantly outperforms the CPLEX solver. The CPLEX solver can solve small problems (group 1) to optimality but fails to solve problems with 3 depots (groups 3 and 4) and even cannot find a feasible solution to most instances in groups 3 and 4 within 2 h. When the number of customers increases (group 2), CPLEX solver also performs poorly as it rarely finds an optimal solution within 2 h. The proposed EBBC algorithm, however, is very fast in solving instances with small number of customers (groups 1 and 3) and is also efficient for instances with more customers but low customer-depot ratio (group 4). When the customer-depot ratio increases (2 depots, 30 customers in group 2), the EBBC algorithm is less effective largely because routing problems can become drastically more complex with more customers to serve. Nevertheless, we observe that the average gap of EBBC solutions for group 2 is only 1.5%, which indicates that EBBC finds high-quality feasible solutions.

We analyze the benefit of each class of cuts. First, the computing time of BBC+ is significantly less than that of BBC for all four groups. This indicates that adding the “additional cuts” (Section 3.3.4) to the master problem can largely reduce the computational complexity of HMCVRP. Note the relative time savings for instances with 20 customers (groups 1 and 3) are more than instances with 30 customers (groups 2 and 4). Second, to see the benefit of the subtour elimination cuts (SEC), we compare the computing time of BBC+SEC to BBC+, and EBBC to BBC+SVC. The results of groups 3 and 4 show clear reduction in computing time by using SEC. Such reduction for group 1 is less significant because problems in group 1 are of small size and BBC+ is already quite efficient. In group 2, however, applying SEC increases the computing time because generating SEC takes significant computational effort (shown in row “SEC”) when the customer-depot ratio becomes higher (15 in group 2, 10 in group 1 and 4, and 6.7 in group 3). These results indicate that a fast procedure for detecting subtour and generating SEC for large problems, if available, can significantly speed up the algorithm. Finally, we consider the single vehicle cuts (SVC) by comparing the results of BBC+SVC to BBC+, and EBBC to BBC+SEC. It is evident that using the single vehicle subproblem (SP3) can greatly reduce the necessity of solving SP2. However, the saving in computing time is not obvious because in these four groups of instances, each depot only has one vehicle. To further study the benefit of SVC, we conduct additional experiments with group 1. We set the vehicle capacity to 200 for the instances in group 1. Let each depot have two vehicles and solve the instances by BBC+ and BBC+SVC. Repeat with each depot having three

Table 6
Performance of different algorithms on instance group 3.

Group 3		BBC	BBC+	BBC+SEC	BBC+SVC	EBBC	SVR
RLB	V	220.9	261.78	274.87	261.75	274.07	n/a
	T(s)	0.23	0.61	0.93	0.57	0.82	
SP1C	N	18393.05	663.3	108.3	692.8	138.45	n/a
	T(s)	277.9	1.9	0.59	1.97	0.75	
SP2C	N	21019	474.55	103.4	6.55	10.15	n/a
	T(s)	150.51	2.82	0.78	0.16	0.18	
SEC	N	n/a	n/a	4155.4	n/a	4430.45	n/a
	T(s)			4.43		5.64	
SVC	N	n/a	n/a	n/a	462.45	113.5	n/a
	T(s)				2.56	0.84	
All-C	N	39412.05	1137.85	4367.1	1161.8	4692.55	n/a
	T(s)	428.41	4.72	5.81	4.68	7.42	
Nodes		245608.95	98230.15	1062.65	105687.35	1251.4	1558650.15
CT(s)		7200	332.77	21.82	381.54	29.07	7200
#OPT		0	20	20	20	20	0
#FS		3	20	20	20	20	0

Table 7
Performance of different algorithms on instance group 4.

Group 4		BBC	BBC+	BBC+SEC	BBC+SVC	EBBC	SVR
RLB	V	295.28	325.18	340.65	325.19	341.32	n/a
	T(s)	0.42	0.97	1.74	0.95	1.74	
SP1C	N	9802.85	1903.95	229.2	2005.9	292.45	n/a
	T(s)	258.87	10.84	7.95	11.68	8.89	
SP2C	N	11742.3	1359.45	280.3	16.65	86.3	n/a
	T(s)	255.24	11.29	5.1	0.54	1.81	
SEC	N	n/a	n/a	16230.65	n/a	16989	n/a
	T(s)			145.54		127.16	
SVC	N	n/a	n/a	n/a	1378.1	232.2	n/a
	T(s)				12.05	5.35	
All-C	N	21545.15	3263.4	16740.15	3400.65	17599.95	n/a
	T(s)	514.11	22.13	158.58	24.27	143.2	
Nodes		146690.05	422188.3	5777.35	370209.4	5167.5	n/a
CT(s)		7200	6337.1	734.03	5871.51	659.12	7200
#OPT		0	5	20	5	20	0
#FS		0	20	20	20	20	0

vehicles. The computational results are showed in [Tables 8 and 9](#), respectively. We can clearly see that the SVC strategy can reduce the computing time of the proposed algorithm.

In addition, we compare the performance of EBBC and the CPLEX solver on instances based on Solomon problems c101 (groups 5 ~ 8) and rc101 (groups 9 ~ 12). The results also show that the proposed EBBC algorithm significantly outperforms the CPLEX solver. See the [Appendix](#) for computational results.

4.2. Locations of depots

We use the 20 instances in group 1 to study how locations of depots affect their collaboration in HMCVRP. For each instance, we move depots A and B in the opposite directions by the same amount along the line that passes them so that their distance becomes m times of that before relocation. Use u_{AB} and u_{AB}^0 to denote the depots' total travel cost with and without collaboration, respectively. Let $v_{AB} = u_{AB}^0 - u_{AB}$ denote the cost savings. For $m = 0.2, 0.6, 1.0, 1.4, 1.8, 2.2$, we solve the HMCVRP by EBBC and calculate the following metrics: (1) relative cost savings of total travel cost compared to the no-collaboration case, i.e., v_{AB}/u_{AB}^0 (2) number of depots designated as TP, and (3) number of customers designated as TP. [Table 10](#) shows the average of these metrics over 20 instances.

Clearly, as the distance between the depots increases, it becomes less likely that the depots are selected as TPs. The average number of depots as TPs decreases from 1 (out of 2 depots) to 0.85 ($m = 1.4, 1.8$ and 0.9 ($m = 2.2$)). On the other hand, the number of customer TPs rises from 0 to 0.2 ($m = 1.4, 2.2$) and 0.3 ($m = 1.8$). More importantly, the relative cost savings decrease as the depots are located farther from each other. The result suggests that the depots can play a more important role than customer nodes in collaborative route planning. The trend of relative cost savings indicates that the marginal benefit of collaboration can become less significant when the depots are far from one another. Therefore, the company that owns the depots (for example, JD) should carefully decide the locations of its depots or properly choose from existing depots when implementing multi-depot collaborative vehicle routing.

Table 8

Performance of BBC+ and BBC+SVC on instance group 1: Each depot has two vehicles.

		BBC+	BBC+SVC
RLB	V	237.29	238.06
	T(s)	0.63	0.66
SP1C	N	1278.2	1188.75
	T(s)	10.58	9.06
SP2C	N	1541.7	1.65
	T(s)	7.89	0.11
SVC	N		1381.95
	T(s)	n/a	3.5
All-C	N	2819.9	2572.35
	T(s)	18.47	12.67
Nodes		49401.85	47274.1
CT(s)		445.81	386.54
#OPT		20	20
#FS		20	20

Table 9

Performance of BBC+ and BBC+SVC on instance group 1: Each depot has three vehicles.

		BBC+	BBC+SVC
RLB	V	233.13	232.57
	T(s)	0.88	0.73
SP1C	N	1297.45	1244.4
	T(s)	18.92	17.57
SP2C	N	1555.1	4.75
	T(s)	11.82	0.18
SVC	N		1417.8
	T(s)	n/a	3.76
All-C	N	2852.55	2666.95
	T(s)	30.74	21.52
Nodes		54713.05	50469.05
CT(s)		889.87	803.12
#OPT		20	20
#FS		20	20

Table 10The results of group 1 with different m setting.

m	Relative cost savings	#Depot TP	#Customer TP
0.2	28.7%	1	0
0.6	26.9%	1	0
1.0	24.5%	1	0
1.4	23.8%	0.85	0.2
1.8	22.7%	0.85	0.3
2.2	22.1%	0.9	0.2

4.3. Allocation of cost savings

In this section, we study a depot's incentive to collaborate with others. While the total travel cost of all depots is reduced in HMCVRP, none of them would prefer its own travel cost to increase. A depot needs to consider its own travel cost if it operates as an independent service provider, or if it is owned by a parent company but accounts for its own profit and loss. Therefore, it is highly necessary to establish a mechanism to properly allocate the cost savings to the depots so that they have no intention to exit the collaborative group. Taking the game-theoretic perspective, the allocation of total cost savings should belong to the core of the cooperative game in which every depot is a player, and the payoff of any coalition is its total travel cost savings. In this way, all depots will be incentivized to collaborate with each other, i.e., the grand coalition is stable.

We use the 20 instances in group 3 to check three mechanisms: The first is no allocation which means each depot operates as suggested by the optimal solution to HMCVRP. The second is equal division in which case the total cost savings of any coalition is evenly allocated among its members. The third is Shapley value allocation which allocates each depot the amount equal to its Shapley value. For each allocation mechanism, we count the number of instances whose allocation of cost savings is in the core. The result is summarized in Table 11. We see that none of the three simple allocation mechanisms performs well. None of the allocation belongs to the core except in one instance with Shapley value allocation. We further investigate the 20 instances and find that only

Table 11
The number of instances whose allocation belong to the core.

Allocation mechanism	#InCore
No allocation	0
Equal division	0
Shapley value allocation	1

5 instances have a non-empty core. One of the reasons is that the game we consider is not a convex game. That is, the marginal benefit of a depot joining an existing coalition is not necessarily increasing with respect to the coalition. Unfortunately, formulating the conditions for the game to be convex is more complex than the scope of this section.

The results in this experiment imply that the grand coalition is not always stable. If the depots have a choice, some may prefer leaving the grand coalition to form smaller coalitions. If the depots are required to collaborate, then the total travel cost is reduced at the expense of some depots' increased travel costs. In this case, the parent company should consider subsidizing these depots. When the depots operate independently and the core exists, a well-designed allocation mechanism becomes extremely important as it can hold the depots in the grand coalition. Otherwise, the depots should evaluate choices of partners in the collaborative route planning.

5. Conclusions

Collaboration has been proven to be an effective approach to improve the efficiency of transportation by exploiting the synergies among collaborators, which may reduce the travel cost and improve the utilization of resources. This research has proposed and efficiently solved a heterogeneous multi-depot collaborative vehicle routing problem. The proposed Benders-based branch-and-cut algorithm has shown "an" overwhelming computational advantage over the CPLEX solver using the commonly adopted big-M transformation-based reformulation. This research also provides benchmark instances for HMCVRP.

Our numerical study has revealed the effect of the depots' locations on cost savings. If a depot is located near to another one, it would be economically effective to have one of them be a transfer point. In this way, a two-depot vehicle routing problem becomes similar to a single-depot VRP. Cost savings are then possible because the depot of transfer point may need less vehicles.

The allocation of the cost savings among the depots is an important issue. Our numerical experiments indicate that simple allocation mechanisms may not be sufficient to keep every depot in the grand coalition. In practice, if a company owns multiple collaborative depots, the company should pay close attention to ensure fairness among the depots. Some depots may need compensation from the parent company to cover the increased travel costs. In the case where depots are independent logistics service providers, the allocation mechanism will affect each depot's choice of partners to form coalitions. The grand coalition is not necessarily stable.

This research can be easily extended to more general scenarios. For example, if two or more vehicles are allowed to load at a TP and a vehicle is allowed to load at multiple TPs, we can remove constraints (8) and (10) in problem P1 and replace the variable q_i by q_{ik} to explicitly denote the quantity that vehicle k loads at TP i . Consequently, the total transshipment quantity at TP i will become $\sum_k q_{ik}$, and constraints (13)~(15) need to be updated to reflect this change. This modification of P1 increases the number of variables, but the algorithm developed in this paper can still apply with minor changes.

Potential research directions include HMCVRP with time windows and relaxed requirements. For example, if a customer can be serviced by multiple vehicles (not necessarily belonging to the same depot), the vehicle capacity may be better utilized with more collaborations becoming feasible and economically beneficial. Designing effective coordination mechanisms to enhance the participation of collaborations among all depots in an urban environment could be another research direction.

CRedit authorship contribution statement

Qihuan Zhang: Conceptualization, Methodology, Software, Data curation, Writing – original draft, Writing – reviewing and editing. **Ziteng Wang:** Conceptualization, Methodology, Validation, Writing – original draft, Writing – reviewing and editing. **Min Huang:** Conceptualization, Methodology, Resources, Writing – reviewing and editing, Supervision, Project administration, Funding acquisition. **Yang Yu:** Methodology, Software. **Shu-Cherng Fang:** Methodology, Writing – reviewing and editing.

Acknowledgments

This work has been supported by the NSFC Major International (Regional) Joint Research Project Grant No. 71620107003; Liaoning Revitalizing Talent Program No. XLYC1802115, Liaoning Innovative Research Team in University Grant No. LT2016007; Fundamental Research Funds for State Key Laboratory of Synthetical Automation for Process Industries Grant No. 2013ZCX11; and 111 Project 2.0 No. B08015.

Appendix. Additional instances and computational results

We generate additional instances based on Solomon problems “c101” and “rc101”. In “c101”, the 100 customers are divided into 10 clusters (10 customers in each cluster) by their geographic distribution. To ensure an equal number of customers in each cluster are included in every instance, we renumber the customers to create a new problem called “c101n” in the following way:

Table A.1

Performance of EBBC and SVR on instance group 5.

Group 5		EBBC	SVR
RLB	V	325.37	
	T(s)	0.33	n/a
SP1C	N	1251.95	
	T(s)	34.66	n/a
SP2C	N	1.65	
	T(s)	0.04	n/a
SEC	N	1431.4	
	T(s)	191.65	n/a
SVC	N	6647.95	
	T(s)	17.44	n/a
All-C	N	9332.95	
	T(s)	243.79	n/a
Nodes		49286.95	4690250.2
CT(s)		1281.35	6732.73
#OPT		18	2
#FS		20	20

Table A.2

Performance of EBBC and SVR on instance group 6.

Group 6		EBBC	SVR
RLB	V	205.88	
	T(s)	1.51	n/a
SP1C	N	5637.6	
	T(s)	337.29	n/a
SP2C	N	24.65	
	T(s)	0.33	n/a
SEC	N	12084.95	
	T(s)	1983.2	n/a
SVC	N	13802.75	
	T(s)	84.64	n/a
All-C	N	31549.95	
	T(s)	2405.46	n/a
Nodes		45976.25	2027929.1
CT(s)		7200	7200
#OPT		0	0
#FS		11	2

Table A.3

Performance of EBBC and SVR on instance group 7.

Group 7		EBBC	SVR
RLB	V	349.34	
	T(s)	0.87	n/a
SP1C	N	275.6	
	T(s)	6.87	n/a
SP2C	N	63.7	
	T(s)	0.76	n/a
SEC	N	4044.45	
	T(s)	66.62	n/a
SVC	N	667.15	
	T(s)	6.84	n/a
All-C	N	5050.9	
	T(s)	81.09	n/a
Nodes		9292.75	2000029.25
CT(s)		392.67	7200
#OPT		20	0
#FS		20	0

Table A.4
Performance of EBBC and SVR on instance group 8.

Group 8		EBBC	SVR
RLB	V	235.29	
	T(s)	2.13	n/a
SP1C	N	906.85	
	T(s)	64.12	n/a
SP2C	N	377.75	
	T(s)	12.76	n/a
SEC	N	20662.75	
	T(s)	1327.1	n/a
SVC	N	1257.25	
	T(s)	23.8	n/a
All-C	N	23204.6	
	T(s)	1427.77	n/a
Nodes		35075.85	1560823.55
CT(s)		5401.32	7200
#OPT		8	0
#FS		20	0

Table A.5
Performance of EBBC and SVR on instance group 9.

Group 9		EBBC	SVR
RLB	V	322.79	
	T(s)	0.32	n/a
SP1C	N	1696.2	
	T(s)	28.88	n/a
SP2C	N	3.7	
	T(s)	0.05	n/a
SEC	N	1966.4	
	T(s)	223.59	n/a
SVC	N	6573.8	
	T(s)	17.84	n/a
All-C	N	10240.1	
	T(s)	270.36	n/a
Nodes		53626.55	2991209.15
CT(s)		1442.22	5927.29
#OPT		19	5
#FS		20	20

Table A.6
Performance of EBBC and SVR on instance group 10.

Group 10		EBBC	SVR
RLB	V	327	
	T(s)	1.58	n/a
SP1C	N	6856.1	
	T(s)	353.95	n/a
SP2C	N	18.5	
	T(s)	0.38	n/a
SEC	N	12433.6	
	T(s)	1838.84	n/a
SVC	N	15755.75	
	T(s)	98.12	n/a
All-C	N	35063.95	
	T(s)	2291.29	n/a
Nodes		48820	739255
CT(s)		7200	7200
#OPT		0	0
#FS		18	4

The customer $10(p-1)+q$ in c101n, $p, q = 1, 2, \dots, 10$, is the p th customer in cluster q in c101. We set the locations of depots to (25, 50) and (55, 50), respectively, in the two-depot instances, and (30, 30), (60, 30) and (45, 60) for the three-depot case. We generate four groups of instances (groups 5 ~ 8) based on c101n in the same way as r101-based instances.

In “rc101”, the customers belong to two categories. Fifty customers are distributed in five clusters (10 customers per cluster). The other fifty are randomly distributed in the map. We regroup the customers in the first category to create a new problem “rc101c” in the same way as c101n. The randomly distributed customers form a new problem “rc101r”. The locations of depots are set to (20,

Table A.7
Performance of EBBC and SVR on instance group 11.

Group 11		EBBC	SVR
RLB	V	369.52	n/a
	T(s)	0.89	
SP1C	N	142.55	n/a
	T(s)	1.31	
SP2C	N	56.15	n/a
	T(s)	0.64	
SEC	N	4244.9	n/a
	T(s)	7.25	
SVC	N	126.5	n/a
	T(s)	1.41	
All-C	N	4570.1	n/a
	T(s)	10.6	
Nodes		1626.1	1963438.85
CT(s)		38.87	7200
#OPT		20	0
#FS		20	0

Table A.8
Performance of EBBC and SVR on instance group 12.

Group 12		EBBC	SVR
RLB	V	346.46	n/a
	T(s)	1.96	
SP1C	N	692.05	n/a
	T(s)	72.38	
SP2C	N	1022.05	n/a
	T(s)	25.66	
SEC	N	18107.95	n/a
	T(s)	783.53	
SVC	N	1138.35	n/a
	T(s)	33.82	
All-C	N	20960.4	n/a
	T(s)	915.39	
Nodes		22728.4	n/a
CT(s)		3491.72	7200
#OPT		12	0
#FS		20	0

35) and (50, 30), respectively, for the two-depot case, and (20, 22), (50, 20) and (35, 50) for three-depot instances. Four groups of instances (groups 9 ~ 12) are generated with equal number of customers being selected from rc101c and rc101r.

We solve each instance in groups 5 ~ 12 by the EBBC algorithm and the CPLEX solver (SVR) for the big-M reformulation of HMCVRP. Tables A.1–A.8 show the performance metrics of EBBC and SVR on instance groups 5 ~ 12, respectively. The results verify our observation that the proposed EBBC algorithm significantly outperforms the CPLEX solver. The CPLEX solver only solves some instances of small problems (groups 5 and 9) to optimality, finds a feasible solution to a few instances of groups 6 and 10, and fails to solve instances including three depots (groups 7, 8, 11 and 12) within 2 h. However, EBBC can efficiently solve the instances with 20 customers and 3 depots (groups 7 and 11) to optimality. EBBC is effective to solve most instances when the customer-depot ratio increases (groups 5, 8, 9 and 12) while some instances are not optimally solved within 2 h. When customer-depot ratio is 15 (groups 6 and 10) the EBBC can find a feasible solution to 73% of the instances. We notice that the customers' clustering distribution will increase the computational difficulty of HMCVRP. The EBBC can solve 90% of the r101-based instances, but this ratio is 63.8% and 57.5% for rc101-based instances and c101-based instances, respectively.

References

- Adulyasak, Y., Cordeau, J.F., Jans, R., 2014. Formulations and branch-and-cut algorithms for multivehicle production and inventory routing problems. *INFORMS J. Comput.* 26 (1), 103–120. <http://dx.doi.org/10.1287/ijoc.2013.0550>.
- Adulyasak, Y., Cordeau, J.F., Jans, R., 2015. Benders decomposition for production routing under demand uncertainty. *Oper. Res.* 63 (4), 851–867. <http://dx.doi.org/10.1287/opre.2015.1401>.
- Agarwal, R., Ergun, Ö., 2008. Ship scheduling and network design for cargo routing in liner shipping. *Transp. Sci.* 42 (2), 175–196. <http://dx.doi.org/10.1287/trsc.1070.0205>.
- Archetti, C., Bertazzi, L., Laporte, G., Speranza, M.G., 2007. A branch-and-cut algorithm for a vendor-managed inventory-routing problem. *Transp. Sci.* 41 (3), 382–391. <http://dx.doi.org/10.1287/trsc.1060.0188>.
- Bai, L., Rubin, P.A., 2009. Combinatorial Benders cuts for the minimum tollbooth problem. *Oper. Res.* 57 (6), 1510–1522. <http://dx.doi.org/10.1287/opre.1090.0694>.
- Benders, J.F., 1962. Partitioning procedures for solving mixed-variables programming problems. *Numer. Math.* 4 (1), 238–252. <http://dx.doi.org/10.1007/bf01386316>.

- Berger, S., Bierwirth, C., 2010. Solutions to the request reassignment problem in collaborative carrier networks. *Transp. Res. E* 46 (5), 627–638. <http://dx.doi.org/10.1016/j.tre.2009.12.006>.
- Bergmann, F.M., Wagner, S.M., Winkenbach, M., 2020. Integrating first-mile pickup and last-mile delivery on shared vehicle routes for efficient urban e-commerce distribution. *Transp. Res. B* 131, 26–62. <http://dx.doi.org/10.1016/j.trb.2019.09.013>.
- Bruglieri, M., Mancini, S., Peruzzini, R., Pisacane, O., 2021. The multi-period multi-trip container drayage problem with release and due dates. *Comput. Oper. Res.* 125, 105102. <http://dx.doi.org/10.1016/j.cor.2020.105102>.
- Buijs, P., Alvarez, J.A.L., Veenstra, M., Roodbergen, K.J., 2016. Improved collaborative transport planning at Dutch logistics service provider Fritom. *Interfaces* 46 (2), 119–132. <http://dx.doi.org/10.1287/inte.2015.0838>.
- Çalik, H., Fortz, B., 2019. A benders decomposition method for locating stations in a one-way electric car sharing system under demand uncertainty. *Transp. Res. B* 125, 121–150. <http://dx.doi.org/10.1016/j.trb.2019.05.004>.
- Chen, J., Jia, S., Wang, S., Liu, Z., 2018. Subloop-based reversal of port rotation directions for container liner shipping network alteration. *Transp. Res. B* 118, 336–361. <http://dx.doi.org/10.1016/j.trb.2018.10.016>.
- Chen, J.H., Lee, D.H., Cao, J.X., 2012. A combinatorial Benders' cuts algorithm for the quayside operation problem at container terminals. *Transp. Res. E* 48 (1), 266–275. <http://dx.doi.org/10.1016/j.tre.2011.06.004>.
- Codato, G., Fischetti, M., 2006. Combinatorial Benders' cuts for mixed-integer linear programming. *Oper. Res.* 54 (4), 756–766. <http://dx.doi.org/10.1287/opre.1060.0286>.
- Cortés, C.E., Matamala, M., Contardo, C., 2010. The pickup and delivery problem with transfers: formulation and a branch-and-cut solution method. *European J. Oper. Res.* 200 (3), 711–724. <http://dx.doi.org/10.1016/j.ejor.2009.01.022>.
- Côté, J.F., Dell'Amico, M., Iori, M., 2014. Combinatorial Benders' cuts for the strip packing problem. *Oper. Res.* 62 (3), 643–661. <http://dx.doi.org/10.1287/opre.2013.1248>.
- Côté, J.F., Haouari, M., Iori, M., 2021. Combinatorial Benders decomposition for the two-dimensional bin packing problem. *INFORMS J. Comput.* <http://dx.doi.org/10.1287/ijoc.2020.1014>.
- Crujssens, F., Bräysy, O., Dullaert, W., Fleuren, H., Salomon, M., 2007a. Joint route planning under varying market conditions. *Int. J. Phys. Distrib. Logist. Manage.* 37 (4), 287–304. <http://dx.doi.org/10.1108/09600030710752514>.
- Crujssens, F., Cools, M., Dullaert, W., 2007b. Horizontal cooperation in logistics: opportunities and impediments. *Transp. Res. E* 43 (2), 129–142. <http://dx.doi.org/10.1016/j.tre.2005.09.007>.
- Elçi, Ö., Noyan, N., 2018. A chance-constrained two-stage stochastic programming model for humanitarian relief network design. *Transp. Res. B* 108, 55–83. <http://dx.doi.org/10.1016/j.trb.2017.12.002>.
- Ergun, O., Kuyzu, G., Savelsbergh, M., 2007. Reducing truckload transportation costs through collaboration. *Transp. Sci.* 41 (2), 206–221. <http://dx.doi.org/10.1287/trsc.1060.0169>.
- Gansterer, M., Hartl, R.F., 2016. Request evaluation strategies for carriers in auction-based collaborations. *OR Spectrum* 38 (1), 3–23. <http://dx.doi.org/10.1007/s00291-015-0411-1>.
- Gansterer, M., Hartl, R.F., 2018. Collaborative vehicle routing: a survey. *European J. Oper. Res.* 268 (1), 1–12. <http://dx.doi.org/10.1016/j.ejor.2017.10.023>.
- Gansterer, M., Hartl, R.F., 2020. Shared resources in collaborative vehicle routing. *Top Off. J. Spanish Soc. Stat. Oper. Res.* 28 (1), 1–20. <http://dx.doi.org/10.1007/s11750-020-00541-6>.
- Hernández, S., Peeta, S., Kalafatas, G., 2011. A less-than-truckload carrier collaboration planning problem under dynamic capacities. *Transp. Res. E* 47 (6), 933–946. <http://dx.doi.org/10.1016/j.tre.2011.03.001>.
- Krajewski, M.A., Kopfer, H., 2006. Collaborating freight forwarding enterprises. *OR Spectrum* 28 (3), 301–317. <http://dx.doi.org/10.1007/s00291-005-0031-2>.
- Li, Y., Chen, H., Prins, C., 2016. Adaptive large neighborhood search for the pickup and delivery problem with time windows, profits, and reserved requests. *Euro. J. Oper. Res.* 252 (1), 27–38. <http://dx.doi.org/10.1016/j.ejor.2015.12.032>.
- Li, J., Rong, G., Feng, Y., 2015. Request selection and exchange approach for carrier collaboration based on auction of a single request. *Transp. Res. E* 84, 23–39. <http://dx.doi.org/10.1016/j.tre.2015.09.010>.
- Lin, C.K.Y., 2008. A cooperative strategy for a vehicle routing problem with pickup and delivery time windows. *Comput. Ind. Eng.* 55 (4), 766–782. <http://dx.doi.org/10.1016/j.cie.2008.03.001>.
- Liu, R., Jiang, Z., Fung, R.Y., Chen, F., Liu, X., 2010a. Two-phase heuristic algorithms for full truckloads multi-depot capacitated vehicle routing problem in carrier collaboration. *Comput. Oper. Res.* 37 (5), 950–959. <http://dx.doi.org/10.1016/j.cor.2009.08.002>.
- Liu, R., Jiang, Z., Liu, X., Chen, F., 2010b. Task selection and routing problems in collaborative truckload transportation. *Transp. Res. E* 46 (6), 1071–1085. <http://dx.doi.org/10.1016/j.tre.2010.05.003>.
- Mancini, S., Ciavotta, M., Meloni, C., 2021a. The multiple multidimensional knapsack with family-split penalties. *European J. Oper. Res.* 289 (3), 987–998. <http://dx.doi.org/10.1016/j.ejor.2019.07.052>.
- Mancini, S., Gansterer, M., Hartl, R.F., 2021b. The collaborative consistent vehicle routing problem with workload balance. *European J. Oper. Res.* 293 (3), 955–965. <http://dx.doi.org/10.1016/j.ejor.2020.12.064>.
- McDaniel, D., Devine, M., 1977. A modified Benders' partitioning algorithm for mixed integer programming. *Manage. Sci.* 24 (3), 312–319. <http://dx.doi.org/10.1287/mnsc.24.3.312>.
- Naoum-Sawaya, J., Elhedhli, S., 2013. An interior-point benders based branch-and-cut algorithm for mixed integer programs. *Ann. Oper. Res.* 210 (1), 33–55. <http://dx.doi.org/10.1007/s10479-010-0806-y>.
- Pan, S., Trentesaux, D., Ballot, E., Huang, G.Q., 2019. Horizontal collaborative transport: survey of solutions and practical implementation issues. *Int. J. Prod. Res.* 57 (15–16), 5340–5361. <http://dx.doi.org/10.1080/00207543.2019.1574040>.
- Pekeć, A., Rothkopf, M.H., 2003. Combinatorial auction design. *Manage. Sci.* 49 (11), 1485–1503. <http://dx.doi.org/10.1287/mnsc.49.11.1485.20585>.
- Shen, K., Huang, J., Wang, Y., 2020. China express-delivery blue book. Fitch Ratings from <https://your.fitchratings.com/china-express-delivery-bluebook.html>.
- Solomon, M.M., 1987. Algorithms for the vehicle routing and scheduling problems with time window constraints. *Oper. Res.* 35 (2), 254–265. <http://dx.doi.org/10.1287/opre.35.2.254>.
- Soysal, M., Bloemhof-Ruwaard, J.M., Haijema, R., van der Vorst, J.G.A.J., 2018. Modeling a green inventory routing problem for perishable products with horizontal collaboration. *Comput. Oper. Res.* 89, 168–182. <http://dx.doi.org/10.1016/j.cor.2016.02.003>.
- Sprenger, R., Mönich, L., 2012. A methodology to solve large-scale cooperative transportation planning problems. *European J. Oper. Res.* 223 (3), 626–636. <http://dx.doi.org/10.1016/j.ejor.2012.07.021>.
- Stoer, M., Wagner, F., 1997. A simple min-cut algorithm. *J. ACM* 44 (4), 585–591. <http://dx.doi.org/10.1145/263867.263872>.
- Taşkın, Z.C., Cevik, M., 2013. Combinatorial benders cuts for decomposing IMRT fluence maps using rectangular apertures. *Comput. Oper. Res.* 40 (9), 2178–2186. <http://dx.doi.org/10.1016/j.cor.2011.07.005>.
- Taherkhani, G., Alumur, S.A., Hosseini, M., 2020. Benders decomposition for the profit maximizing capacitated hub location problem with multiple demand classes. *Transp. Sci.* 54 (6), 1446–1470. <http://dx.doi.org/10.1287/trsc.2020.1003>.
- Verdonck, L., Caris, A.N., Ramaekers, K., Janssens, G.K., 2013. Collaborative logistics from the perspective of road transportation companies. *Transp. Rev.* 33 (6), 700–719. <http://dx.doi.org/10.1080/01441647.2013.853706>.
- Verstichel, J., Kinable, J., Causmaecker, P.De., Bergh, G.V., 2015. A combinatorial benders decomposition for the lock scheduling problem. *Comput. Oper. Res.* 54, 117–128. <http://dx.doi.org/10.1016/j.cor.2014.09.007>.

- Wang, Y., Zhang, D., Liu, Q., Shen, F., Lee, L.H., 2016. Towards enhancing the last-mile delivery: an effective crowd-tasking model with scalable solutions. *Transp. Res. E* 93, 279–293. <http://dx.doi.org/10.1016/j.tre.2016.06.002>.
- Xu, S.X., Huang, G.Q., Cheng, M., 2017. Truthful, budget-balanced bundle double auctions for carrier collaboration. *Transp. Sci.* 51 (4), 1365–1386. <http://dx.doi.org/10.1287/trsc.2016.0694>.