Omega 108 (2022) 102582

Contents lists available at ScienceDirect

Omega

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

Bundle generation for last-mile delivery with occasional drivers[☆]

Simona Mancini[a,b,*], Margaretha Gansterer[a]

[a] University of Klagenfurt, Department of Operations, Energy, and Environmental Management, Universitätsstraße 65-67, Klagenfurt 9020, Austria
[b] University of Eastern Piedmont, Department of Science, Innovation and Technology, Viale Teresa Michel 11, Alessandria 15121, Italy

ARTICLE INFO

Article history:
Received 17 June 2021
Accepted 5 December 2021
Available online 9 December 2021

Keywords:
Routing
Occasional drivers
Last-mile delivery
Matheuristic

ABSTRACT

In this paper, we present the vehicle routing problem (VRP) with occasional drivers (OD) and order bundles (OB). The problem VRP-OD-OB is an extension of the VRP-OD, where instead of assigning one customer per driver, drivers are assigned bundles of customers. To deal with the bundle-to-driver assignment, a bidding system is exploited, in which a company offers a set of bundles and the drivers raise their bids. These bids depend on features such as the drivers' destination, flexibility in deviating from the shortest path, and willingness to offer service. To generate valuable bundles of customers, we propose two strategies: (i) an innovative approach based on the creation of corridors, and (ii) a traditional approach based on clustering. Through an experimental study, carried out on randomly generated instances and on a real road network, we show that the innovative corridor-based approach strongly outperforms the clustering-based approach. Given a set of bundles and a corresponding set of bids, we provide a mathematical formulation and valid inequalities to solve the VRP-OD-OB. To address larger instances, we design an efficient large neighborhood search-based matheuristic. The results of an extensive computational study show that this method provides near-optimal solutions within very short run times. An analysis of the impact of drivers' flexibility and willingness levels on the percentage of customers assigned to ODs is presented. Moreover, the case in which ODs dynamically appear at regular time intervals is investigated. Also in this dynamic setting, considerable total cost reductions are shown. Moreover, we derive several important managerial insights, which include the observation that it is not necessary to provide a high number of bundles to achieve good quality solutions. Companies should rather focus on generating fewer but more attractive bundles.

1. Introduction

E-commerce has become increasingly popular in the last decade. Global e-commerce sales surged to $29 trillion in 2019 [1] and has been experiencing exponential growth since then. The advent of the SARS-CoV-2 pandemic further increased the number of online purchasers, with a large number of users who were previously unfamiliar with online purchasing adopting this system. Several years ago, only a few retailers, such as Amazon and Zalando, offered online purchasing, and the online marketplace was shared among them. Then, other big brands, such as North Sails, Decathlon, and Cisalfa, also launched private purchasing websites. Nowadays, every medium or large retailer is almost obligated to offer online purchasing to stay in the market. In fact, as pointed out in [2], single-brand retailers offer a more hedonic experience to the customer, and, through dedicated promotions, it is easier to increase customers' fidelity. Furthermore, customers' reviews and stars ratings help to attract new potential customers who did not have direct experience with the brand yet, as stated in [3]. The adoption of an owned online-purchasing system, can be advantageous, under specific market conditions, also for companies operating in the building materials sector, as pointed out in [4], where the case of a brick-and-mortar retail stores is analyzed.

E-commerce is attractive because users can compare thousands of alternatives, purchase at any time of the day from their laptop, tablet, smartphone, or even smartwatch, and have the goods delivered directly to their home. However, e-commerce results in a huge increment in the costs for last-mile delivery. In fact, although online retail can strongly contribute to increase the number of potential customers, this may require an enhancement of the delivery fleet, in order to fulfill such a large number of or-

☆ This manuscript was processed by Associate Editor Kis.

* Corresponding author at: University of Klagenfurt, Department of Operations, Energy, and Environmental Management, Universitätsstraße 65-67, 9020 Klagenfurt, Austria.

E-mail addresses: simona.mancini@aau.at (S. Mancini), margaretha.gansterer@aau.at (M. Gansterer).

https://doi.org/10.1016/j.omega.2021.102582

ders, which requires a large investment [5]. While large companies such as Amazon already have their own efficient distribution network, medium-sized retailers face very high fulfillment costs. This engendered the concept of crowdshipping, which is an innovative delivery system in which in-store purchasers deliver orders to customers. Crowdshippers, more commonly known as occasional drivers (ODs), are generally much cheaper than traditional drivers. This is because they are paid only if they are assigned a delivery. Their compensation typically depends on the length of the detour required for the delivery. The detour is generally computed as the difference between the shortest path from the depot to the OD's final destination, visiting all the assigned customers, and the direct path from the depot to the OD's final destination. If an OD is assigned orders with negligible detours, the compensation can be low. Conversely, while traditional drivers are always available, in accordance with their working contracts, the availability of ODs is less predictable. Furthermore, they are not obliged to accept a task if it is not convenient for them. Crowdshippers are not only useful in e-commerce, but they can be efficiently exploited also in the meal-delivery context, as discussed in [6].

As pointed out by Ulmer and Savelsbergh [7], to prevent or mitigate the negative effects associated with the uncertainty of ODs' availability, it is convenient for companies to integrate ODs with a fixed owned fleet. However, efficiently planning and performing this integration may prove to be an extremely challenging issue. In [8], the authors propose the integration of in-house drivers and crowdshippers. The latter are full- or part-time contractual, but non-professional, drivers. According to this assumption, the nature of the problem becomes static and deterministic since the availability of ODs for a specific day is supposed to be known in advance. Clearly, if crowdshippers' availabilities are hard to predict and orders arrive on short notice, dynamic and stochastic routing problems will have to be tackled. However, there are real-world applications where the delivery plan must be formulated in advance. In particular, if in-house fleets must leave in the morning for performing deliveries far away from the depot, waiting for potential ODs is not advisable. Therefore, it is necessary to know in advance whether to load customers' goods on in-house vehicles or assign them to ODs. Examples of this include retailers who do not offer same-day delivery services. In these cases, a static and deterministic model is appropriate. A dynamic representation would work, for instance, in the context of electrical/electronic material distribution (such as cables), for which a large stock is available in the warehouse. In this case, all the required materials can be loaded in-house vehicles. Once a suitable OD appears, a new cable can be given to them, and the traditional driver can skip the delivery. Nevertheless, in cases where customer requests are completely personalized and not interchangeable, such a strategy cannot be applied. Therefore, decisions on assigning customers to owned fleets or ODs must be taken in advance and cannot be changed afterwards. These cases can be modeled as static and deterministic problems, in which an OD's availability must be known a priori.

In this context, we introduce the vehicle routing problem (VRP) with occasional drivers (ODs) and bundles of orders (VRP-OD-OB). It considers the possibility of assigning bundles of customers to drivers based on a bidding system. We assume that submitted bids depend on the detour required to serve the customers included in the bundle and on the OD's level of willingness to perform the deliveries. A mathematical formulation and valid inequalities are proposed. The system tackles the problem of (i) generating, and (ii) assigning bundles, and (iii) suggests which customers should be served by the fleet of company-owned vehicles. Since the number of feasible bundles grows exponentially with the size of the problem, only a subset of bundles can be offered for bidding. An important contribution of our study is that we propose an innovative bundle-generation technique based on geographical corridors. We compare this approach to a more traditional clustering method. Extensive computational experiments show that the newly proposed corridor-based approach strongly outperforms the more traditional clustering-based one. Even in the case of dynamically appearing ODs, considerable total cost reductions can be obtained. From methodological point of view, we propose a large neighborhood search (LNS)-based matheuristic (MH). We can show that this method performs well, obtaining near-optimal solutions (less than 1% from the best known solution) in a short computational time. Finally, we provide managerial insights on the impact of drivers' characteristics on the ratios of customers served by ODs.

All instances have been made publicly available [9] to encourage other researchers to contribute to this highly relevant and dynamically evolving field.

The rest of this paper is organized as follows: A literature review is reported in Section 2. Section 3 provides a detailed description of the newly introduced problem. A mixed-integer programming model is presented in Section 4. Section 5 deals with the generation of attractive bundles. The bidding problem is addressed in Section 6, while Section 7 is devoted to the description of the large neighborhood search (LNS)-based matheuristic (MH), which we have proposed for efficiently handling large-sized instances. Computational experiments are presented in Section 8. Finally, the conclusions and future developments are discussed in Section 9.

2. Literature review

A recent survey by Archetti and Bertazzi [10] identifies crowdshipping to be among the main routes that must be considered by future research in the field of e-commerce and last-mile delivery. Although crowdshipping became popular only recently, several relevant studies on it can be found in the literature. Most of them address the topic from an economic perspective. In these papers, the term "crowdshipping" is extended to private drivers willing to perform deliveries but does not necessarily refer to in-store customers who undertake orders when leaving the store. Le and Ukkusuri [11] investigated the factors that can influence retailers' choice to use crowdshipping. An empirical analysis of real crowdshipping systems adopted in the US was presented by Ermagun et al. [12]. The potential impact of crowdshipping on traffic and vehicle emissions was studied by Simoni et al. [13]. The authors used a simulation-based approach to address the case of Rome. For a survey on crowdsourced delivery we refer interested readers to [14].

In [15], the authors address the case of a platform which dynamically receives transportation requests and tries to find a feasible match with companies willing to perform extra tasks to be integrated in their own delivery plan. However, even if this paper refers to crowdshipping, it involves companies already performing tasks rather than ODs. A very similar setting is addressed in [16], in which, differently from Allahviranloo and Baghestani [15], a static version of the problem is considered, where all transportation requests are known in advance. In this paper, we focus on the literature that describes crowdshipping problems involving in-store customers' willingness to perform deliveries after leaving the store. The seminal paper by Archetti et al. [17] was the first to introduce the VRP-OD. The authors assume that at most one customer can be assigned to a given OD and that the compensation is constant and does not depend on the required detour. The authors presented an integer programming formulation for the VRP-OD and designed a multi-start heuristic approach that combined a tabu search and a variable neighborhood search. Macrina et al. [18] extended the work of [17] by considering delivery time windows and allowing multiple customer assignments to ODs. In contrast to our work, they did not consider a bidding system but calculated the drivers' compensation based on the required detour. Therefore, they did not consider features such as the OD's flexibility or their willingness to perform deliveries.

Operational time windows and transshipment nodes, an extension of the VRP-OD, have been proposed in [19] and [20]. A crowdsourced dynamic pickup-and-delivery system was studied by Arslan et al. [21], while [22] addressed a stochastic and dynamic problem involving in-store customers as ODs. In [23], the authors proposed a sustainable crowdsourced delivery system in which the ODs perform their deliveries using public transportation; the authors also assessed the economic and environmental impacts of such a system. A similar system was addressed by Gatta et al. [24], where crowdshippers perform deliveries to automated parcel lockers using public transportation. Another sustainable crowdshipping system was proposed by Lin et al. [25], in which bicycles are used to perform OD deliveries. In [26] a mixed delivery system is considered, where cyclists and pedestrians are integrated with the own fleet in order to perform deliveries at the minimum cost. Crowdshippers submit bids that can be accepted or rejected by the company. Those, who are selected for delivery, meet the truck at fixed stations where they collect the delivery parcels.

The first work to consider heterogeneous ODs, characterized by different levels of flexibility, was by Behrend and Meisel [27]. However, they did not consider an actual bidding system, using only flexibility to determine the set of all feasible customer tuples to be assigned to ODs. An exact method for the same problem was proposed by Behrend et al. [28]. Dahle et al. [29] were the first to consider the possibility of refusing an assignment if the compensation offered is too low. This issue was not addressed within a bidding system in which drivers submit bids that are convenient for them; however, it did consider a minimum compensation threshold for each OD. Furthermore, compensations are defined a priori by the company. In [30] the authors propose to exploit employees of their distribution centers for crowdshipping online orders on their way back from work. Employees communicate a minimum expected earning per time unit and a maximum acceptable driving time. The company decides which tasks to assign to each employee in order to maximize the number of tasks performed by the employees. This objective comes from the idea that subcontracting deliveries to employees is cheaper than performing them. Therefore, the lower the number of deliveries carried out by the company, the greater the gain. However, this is not always the case. While exploiting ODs for reaching customers far from the distribution center would yield a clear advantage for the company, it could be cheaper to serve nearer customers with the fleet owned by the company. Therefore, we decided to consider a system in which the goal is to minimize the total delivery cost exploiting a mixed distribution system (owned fleet plus ODs). Our experiments show that the optimal solution almost always involves both types of delivery and therefore more tasks assigned to ODs do not necessarily lead to lower total costs.

In [31] the authors propose a system in which a centralized platform generates personalized bundles of requests (named menus) for each driver. Drivers do not bid for bundles but just communicate, for each menu, if they are willing or not to fulfill it. After receiving feedback from all the drivers, the platform performs the menus-to-drivers assignments. Flexibility and willingness to work impact drivers' choices to accept or to reject a menu. The authors model the problem as a bi-level Stackelberg game. A very similar setting is addressed in [32], where the authors solve a more general problem, in which a platform is used to match customers' requests with suppliers. This system is not only specifically suited for transportation requests, but can be applied to any type of service requests.

As pointed out in [33], transportation orders should be offered in bundles, not individually. This is based on the *subadditivity* of costs, i.e., the fact that serving a bundle of requests might result in a lower total cost than the sum of the costs of serving all orders individually. Hence, the process of order-to-driver assignment should be modeled as a combinatorial optimization problem, where the values of bundles, rather than individual orders, are considered. In contrast to the collaborative routing problem addressed in [33], where carriers identify a priori the set of customers to be offered in the auction, we assume that the set of customers to be assigned to ODs is not predefined. Hence, each customer may be served by ODs or by company-owned vehicles. Thus, the auction problem must be addressed along with the owned-fleet routing problem. To solve this problem, we propose a mathematical model that (i) tackles customers in OD assignments, (ii) provides a routing plan for the company fleet, and (iii) assigns bundles to the ODs.

In order to highlight our contribution to the existing literature, we report, in Table 1, the list of features addressed by each relevant paper in the field. As it can be evinced from the table, our study clearly fills a gap in the existing literature. To the best of our knowledge, we are the first to consider a VRP-OD involving a bundling and a bidding system, drivers' flexibility and willingness to work, and dynamic aspects, simultaneously.

3. Problem description

In this section, we define the VRP-OD-OB. A company has to fulfill a set of orders (I), where customer i requests a given quantity q_i. All deliveries start from a common depot (0). Let us identify the set of nodes involved in the network, i.e., the customers (related to an order) and the depot 0, as $I_0 = I \cup 0$. For each pair of nodes (i and j) in I_0, we assume a travel distance d_{ij} and a travel cost c_{ij}, which are known in advance. For each customer, the company can choose between two options: (i) serving the customer with its own fleet or (ii) assigning them to an OD who will perform the delivery. The owned fleet is composed of m identical vehicles with a capacity equal to $Q_{\max}$. Each vehicle in the owned fleet starts from the depot and must return to it. A given set of ODs (Ω) is available. We assume that the ODs start from the depot and fulfill a set of deliveries, i.e., serve a set of customers, on the way back to their homes (generally, their destinations). The capacity of an OD's vehicle is indicated by $Q^{OD}_{\max}$.

The company receives a set K of bids; each of these bids is associated with a bundle of customers (τ_k). For each bid, the individual value b_k for serving the bundle τ_k and the offering driver o_k are known. We assume that each OD can submit bids for an unlimited number of bundles but that, at most, one bid per OD can be accepted by the company. The same assumption is made in several other studies, such as [33] and [34], that deal with auction-based collaborative transportation. This way, the exposure problem – which is well known in auction theory, [35] – can be avoided. This prevents situations where an OD wins several bundles but does not have enough capacity to fulfill them.

If a bid is accepted, all customers belonging to it are assigned to the corresponding OD. A customer cannot be assigned to more than one driver; hence, all the bundles assigned to ODs must be disjointed. The goal is to minimize the total cost, which is the sum of the routing costs of the owned vehicles and the costs associated with the accepted bids.

Summarizing, our optimization approach can be described as a three-steps system:

1. The company generates potentially attractive bundles without having information about the ODs, but exploiting only spatial relationship among customers' locations.

Table 1
Overview of features addressed in the literature.

	Multiple cust.	Compensation for detour	Flexibility	Willingness	Bidding	Mixed system	Dynamic
Archetti et al. [17]	No	No	No	No	No	Yes	No
Macrina et al. [18]	Yes	Yes	No	No	No	Yes	No
Macrina and Guerriero [19]	Yes	Yes	No	No	No	Yes	No
Macrina et al. [20]	Yes	Yes	No	No	No	Yes	No
Arslan et al. [21]	Yes	Yes	Yes	No	No	No	Yes
Dayarian and Savelsbergh [22]	Yes	Yes	Yes	No	No	No	Yes
Kafle et al. [26]	Yes	Yes	Yes	Yes	Yes	Yes	No
Behrend and Meisel [27]	Yes	Yes	Yes	Yes	No	No	No
Dahle et al. [29]	Yes	Yes	Yes	Yes	No	Yes	No
Lin et al. [25]	Yes	Yes	Yes	Yes	No	No	No
Boysen et al. [30]	Yes	Yes	Yes	Yes	No	No	No
Horner et al. [31]	Yes	Yes	Yes	Yes	No	No	No
Mofidi and Pazour [32]	Yes	Yes	Yes	Yes	No	No	No
Our paper	**Yes**	**Yes**	**Yes**	**Yes**	**Yes**	**Yes**	**Yes**

2. ODs assess the offered bundles and make a bid for attractive ones. We consider that the bid depends on the OD's actual detour, her flexibility and her willingness to work.
3. After the company has received all bids, (i) it decides which bids to accept, and (ii) the routing plan for the owned fleet to serve customers, which are not included in the accepted bids.

4. Mathematical formulation

The VRP-OD-OB can be formulated as follows:

I	set of customers
I_0	set of nodes involved in the network (customers and depot)
Ω	set of occasional drivers
K	set of bids
m	number of available vehicles
c_{ij}	travel cost between node i and node j
q_i	demand of customer i
$Q_{\max}$	capacity of the owned vehicles
$Q^{OD}_{\max}$	capacity of the ODs' vehicles
b_k	price of bid k offered by an OD
τ_k	bundle related to bid k
o_k	OD who submitted bid k
C_k	set of customers belonging to bundle τ_k
Z_i	binary variable that takes value 1 if customer i is visited by a company-owned vehicle and takes 0 otherwise
X_{ij}	binary variable that takes value 1 if node j is visited by a company-owned vehicle just after node i and takes 0 otherwise
Y_k	binary variables that take value 1 if bid k is accepted and take 0 otherwise
Q_i	non-negative variables representing the cumulative load at node i, expressed as the total quantity of demand delivered by a vehicle along its route, when leaving node i

$$\min \sum_{i\in I_0}\sum_{j\in I_0} c_{ij}X_{ij} + \sum_{k\in K} b_k Y_k \tag{1}$$

$$\sum_{j\in I} X_{0j} \le m \tag{2}$$

$$Z_j + \sum_{k\in K|j\in C_k} Y_k = 1 \qquad \forall j \in I \tag{3}$$

$$\sum_{k\in K|o_k=\omega} Y_k \le 1 \qquad \forall \omega \in \Omega \tag{4}$$

$$\sum_{i\in I_0} X_{ij} = Z_j \qquad \forall j \in I \tag{5}$$

$$\sum_{i\in I_0} X_{ij} = \sum_{i\in I_0} X_{ji} \qquad \forall j \in I \tag{6}$$

$$Q_j \ge Q_i + q_j - 2Q_{max}(1 - X_{ij}) \qquad \forall i \in I \ \ \forall j \in I \tag{7}$$

$$0 \le Q_j \le Q_{max} \qquad \forall j \in I \tag{8}$$

$$X_{ij} \in \{0,1\} \qquad \forall i \in I_0 \ \ \forall j \in I_0 \tag{9}$$

$$Y_k \in \{0,1\} \qquad \forall k \in K \tag{10}$$

$$Z_i \in \{0,1\} \qquad \forall i \in I \tag{11}$$

The objective (1) is to minimize the total cost for the company. The number of vehicles exploited by the company cannot exceed the number of available vehicles in the owned fleet, as stated in Constraint (2). Constraints (3) and (4) ensure that each customer is directly served by the company or assigned to one of the ODs and that one bid at most is accepted for each OD. If a customer is served by the owned fleet, they must be visited only once, as stated in Constraint (5). Constraint (6) ensures the continuity of the routes. Constraint (7) tracks the cumulative load at the nodes and implies sub-tour elimination, while Constraint (8) ensures that vehicle capacity is respected. Finally, Constraints (9), (10), and (11) specify variable domains.

To strengthen the formulation, we add the following valid inequalities, for which we involve a new set of variables L_{ij} that represent the load carried on arc (i,j).

$$\sum_{j\in I_0} L_{ji} - \sum_{j\in I_0} L_{ij} = q_i Z_i \qquad \forall i \in I \tag{12}$$

$$\sum_{j\in I_0} L_{j0} - \sum_{j\in I_0} L_{0j} = -\sum_{j\in I} q_j Z_j \tag{13}$$

$$L_{ij} \le Q_{max} X_{ij} \qquad \forall i \in I_0 \ \ \forall j \in I_0 \tag{14}$$

$$L_{i0} = 0 \quad \forall i \in I_0 \tag{15}$$

Constraint (12) states that the quantity delivered to each customer is equal to its demand. Constraint (13) ensures that the total delivered quantity is equal to the sum of the demands of the customers served by the company's fleet. Constraint (14) limits the load carried by a vehicle according to the vehicle capacity $Q_{\max}$. Finally, Constraint (15) forces the vehicles to return to the depot empty.

5. Attractive bundle generation

The literature states that bundles of transportation requests can be built either by sellers or by buyers (see [36] and [37]). However, the number of bundles grows exponentially with the number of customers. Therefore, even small-sized instances may become intractable. To make practical problems tractable, a set of *potentially attractive* bundles should be generated [33]. Identifying profitable bundles is an important issue that can determine the success or failure of the entire distribution system. In this paper, we propose two bundle-generation strategies: (i) an innovative corridor-based approach and (ii) a traditional clustering-based approach.

5.1. Cluster-based bundling

The basic idea is to create spatial-based clusters of customers. These bundles are assumed to be profitable since a single driver can serve all of them with a relatively small marginal cost. In fact, customers belonging to the same cluster are located very near to each other. Therefore, whichever is the final destination of the driver, the additional cost for visiting all the customers in the bundle, compared to visiting only one of them, is very small. Thus, it is convenient to offer clustered customers in the same bundle. Since ODs' vehicles have limited capacity, the number of customers that can belong to the same cluster is limited by the constraint that the demands of all customers must be accommodated in a single vehicle. We developed an IP model to generate exactly N_n clusters, where the maximum intra-cluster distance, i.e., the maximum distance between two customers belonging to the cluster, is minimized. The IP model for cluster-based bundling involves the following additional sets and decision variables:

$N = \{1.N_n\}$	set of clusters
w_{in}	binary variable having value 1 if customer i is assigned to cluster n and 0 otherwise
R_n	non-negative variable representing the maximum intra-cluster distance for cluster n

The problem can be formulated as follows:

$$\min \sum_{n \in N} R_n \tag{16}$$

$$\sum_{n \in N} w_{in} = 1 \quad \forall i \in I \tag{17}$$

$$R_n \geq d_{ij}(w_{in} + w_{jn} - 1) \quad \forall i \in I \ \ \forall j \in I \ \ \forall n \in N \tag{18}$$

$$\sum_{i \in I} q_i w_{in} \leq Q_{max}^{OD} \quad \forall n \in N \tag{19}$$

$$w_{in} \in \{0, 1\} \quad \forall n \in N \ \ \forall i \in I \tag{20}$$

The objective function is given in (16). Constraint (17) ensures that each customer is assigned to only one cluster. Constraint (18) helps identify the maximum intra-cluster distance. The constraints are binding only if customers i and j both belong to cluster n. If there is only one customer between i and j or if none of them belong to n, the constraints impose that R_n is greater than or equal to 0 and -1, respectively. These conditions are always respected, given the nature of the variables. Constraint (19) ensures that the total demand of the customers in a cluster does not exceed the vehicle capacity. The domain of the variables is specified by constraints (20).

The minimum number of clusters $N_{\min}$ needed to obtain a feasible partition of customers without exceeding vehicle-capacity constraints can be computed as $N_{\min} = \lceil \sum_{i \in I} q_i / Q_{\max}^{OD} \rceil$.

To generate potentially attractive bundles, we iteratively ran the model described above for values of N_n varying in the range $[N_{\min}, |I|]$. The solution of the problem for $N_n = |I|$ corresponds to $|I|$ bundles, with each one containing a single customer. The offered bundle set comprises all the clusters generated by this procedure, excluding duplicates.

5.2. Corridor-based approach

It has been mentioned that the ODs' paths start from the depot and that they perform their deliveries on the way back to their destinations. Thus, attractive bundles may contain not only customers that are close to each other but also close to the direct path between the depot and ODs' destinations. In fact, such bundles, even if they contain customers far from each other, imply a very short detour for the driver; therefore, they may be potentially attractive. This kind of bundle cannot be generated with the clustering-based approach. Thus, we developed an additional bundling method that we denote as the corridor-based approach.

The corridor-based approach involves identifying the circular sector centered at the depot. It is defined by the smallest angle α, for which all the customers are included in this sector. This sector is then split into n_s identical small sectors characterized by an angle of size α/n_s. We then explore each small sector separately and consider all the customers included in the sector to be a bundle. If the total demand of these customers exceeds the vehicle capacity ($Q_{\max}^{OD}$), the clustering-based approach is repeated on this subset of customers (denoted as I^s). As previously stated, the minimum number of clusters needed to create feasible bundles is $n_{\min}^s = \lceil \sum_{i \in I^s} q_i / Q_{\max}^{OD} \rceil$.

The corridor-based approach is iteratively repeated with different values of n_s. The bundles obtained – excluding duplicates – constitute the set of offered bundles.

Fig. 1 shows attractive bundles generated with the corridor-based approach. The depot is represented by a red square and customers by blue circles, while the ODs' homes are indicated by green triangles. Attractive bundles are circled in red.

In Fig. 2, we provide a comparison of the most attractive bundles generated by the corridor-based approach (a) and clustering-based approach (b). It can be seen that the bundles generated by (a) are much more attractive, since they imply very small detours for the ODs; the bundles generated by (b) imply much longer detours and, therefore, are less attractive. Note that we do not report the exhaustive list of feasible bundles but only a subset of them that can be found by each approach. This is to compare their *shape*.

Finally, we would like to remark that the optimization model, described in Section 4, takes the list of bundles as input. We propose two rational procedures to generate attractive bundles. However, the system works independently from the rule used to generate bundles; even with completely randomly generated ones. The attractiveness of bundles, however, strongly impacts the total cost. Therefore, it is important to generate attractive bundles.

6. The bidding problem

In OD distribution systems, pricing or bidding is a challenging issue. Archetti et al. [17] do not address an actual bidding phase

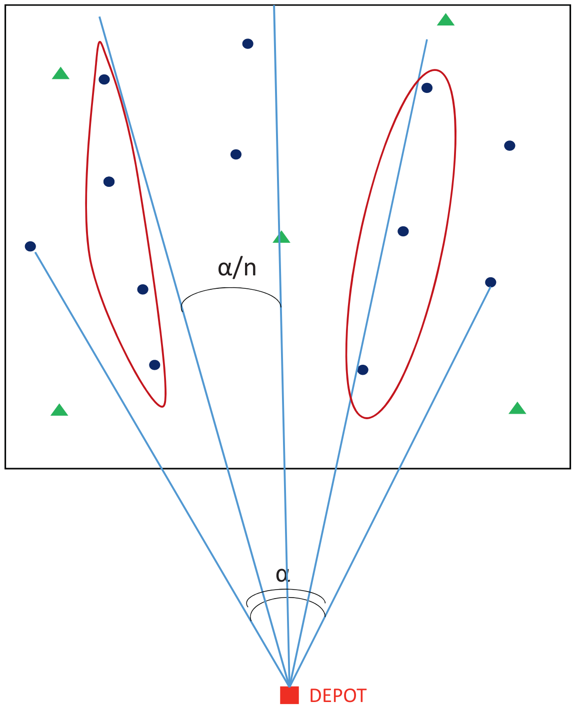

Fig. 1. Attractive bundles generated by the corridor-based approach.

but consider a fixed compensation for each delivery performed, which is independent of the detour implied for the OD. This may be considered unfair as two different ODs may incur completely different extra mileages to serve the same customer but would receive the same compensation. However, the authors argue that to pay a compensation proportional to the actual detour required, it would be necessary to know the home locations of all the ODs. This would expose the system to strategic behavior, as ODs could declare that they live far away from the customers' area to receive a higher compensation. In [29], the authors addressed the pickup-and-delivery VRP with ODs. In contrast to [17], they considered ODs refusing the offer proposed by the company if the compensation is too low. This aspect is modeled considering a minimum compensation threshold for each OD.

While the company decided on the compensation scheme in the previous papers on ODs, in this paper, we let the ODs decide (i) the bundles for which they want to bid and (ii) the value of the bid. In reality, bidding decisions are entirely up to drivers. The company just receives the bids from them without knowing how prices are computed. In our problem, we consider bids as input from ODs, but, in order to have realistic bids, we generate an automatic bidding systems which tries to simulate rational ODs behavior. To fully characterize each OD ω we need to introduce some parameters: (i) *flexibility*, which represents the maximum acceptable detour, computed as the difference between the shortest path from the depot to the OD's final destination, visiting all the customers in the bundle, and the direct path from the depot to the OD's final destination; (ii) *willingness to work* (ϕ_ω), where $\phi_\omega = 1$ describes a neutral behavior, i.e., where the ODs' bids reflect exactly the actual detour implied. In case of a lower willingness (i.e., $\phi_\omega > 1$), the bid prices are increased since the ODs agree to perform a delivery only if they find it very convenient. Values smaller than 1 ($\phi_\omega < 1$) indicate that the driver is willing and, therefore, reduces the bid price to have a greater chance of winning the order. The value of a bid k is calculated as the detour length, δ_k, (i.e. the detour needed by OD, o_k, to serve customers belonging to bundle τ_k), multiplied by a unitary cost c^u, plus a fixed cost c^f, for each customer belonging to the bundle. These are all multiplied by the willingness-to-work parameter (ϕ_{o_k}) associated with the OD who submitted the bid o_k. This is formulated as follows.

$$b_k = (c^u \delta_k + c^f |\tau_k|)\phi_{o_k} \tag{21}$$

Moreover, the flexibility level indirectly impacts the bidding process. In fact, an OD places a bid for a bundle only if the related detour δ_k, is lower than the maximum value allowed ($\delta_{o_k}^{MAX}$). The latter corresponds to the OD's flexibility level.

Thus, we consider compensation schemes depending on the detour required, without forcing the ODs to reveal their home locations to the company. It should be noted that the optimization model presented in Section 4 receives bids as input data. However, this section proposes a mechanism to simulate rational OD's behavior as well as realistic bids. In addition to the presented approaches, other rational ways might exist.

It is worth noting that we are considering bundles that potentially comprise several customers. Therefore, the detour implied by serving a bundle must be computed through solving a special version of a traveling salesman problem (TSP), where the origin and the destination of the tour are the depot and the OD's destination, respectively. Since this problem has to be potentially solved for each pair of OD and bundle, the computational times of the bidding phase are a challenging issue. To reduce these times, we introduce a pre-processing check, where we examine all the customers within a bundle. If the detour imposed by even one of these customers (if served individually) is higher than the OD's flexibility level, no bid can be placed. In this manner, the number of TSPs that needs to be solved for the bidding phase is considerably reduced. Finally, the bid offered by the OD is computed as the compensation associated with the detour, multiplied by the willingness (ϕ).

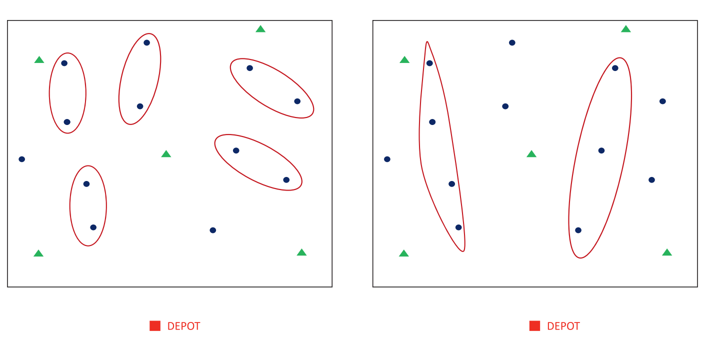

Fig. 2. Comparison of attractive bundles generated by the clustering-based approach (a) and the corridor-based approach (b).

7. Solution approach: Large neighborhood search

LNS is a metaheuristic framework based on the idea that searching large neighborhoods results in finding high-quality solutions by overcoming local minima. The neighborhoods to be explored are implicitly defined by specific *destroy operators*. The partially destroyed solutions are transformed into feasible solutions by means of *repair operators*. LNS has been successfully applied to several routing problems, as reported in [38].

Destroy operators may be defined in different ways. For instance, in routing problems, an operator can destroy k routes and leave the remaining unchanged or remove some arcs belonging to different routes from the current solution. This removal of arcs (or entire routes) can be based on a deterministic or randomized approach. The results from the literature indicate that operators holding a random component show better performances and prevent early convergence toward the local minima [39].

Generally, greedy construction heuristics are used to rebuild the solution. These methods are very fast but not always accurate as only a single solution is analyzed, from among all the feasible solutions that can be generated from the partially destroyed one. Naturally, this may slow down the process of reaching high-quality solutions.

Recently, a hybridized version of LNS, named LNS-based MH, has been successfully proposed in the literature. In this framework, an MIP model, which is run with an execution time limit, is exploited to rebuild the partially destroyed solution. This way, a large portion (often the totality) of the subset of the solutions belonging to the neighborhood can be analyzed. MHs have been successfully applied to several highly constrained routing problems [39,40,41].

In [40], the destroy operators work on arc variable removal from the solution of small subsets of routes, forcing the others to remain unvaried. However, in [39] and [41], the destroy operators do not directly work on arc variables but on customer-to-vehicle assignments.

Working on assignment variables allows the analysis of larger neighborhoods. On the one hand, their exhaustive exploration may require larger computational times since the routing plan must be built from scratch. On the other hand, working on arc variables allows one to start from partially destroyed solutions already containing several arcs; therefore, the rebuild operation is faster, while the size of the addressed neighborhood is smaller. This may negatively affect effectiveness. Finding a balanced compromise between efficiency and effectiveness is a challenging task. In this paper, we propose destroy operators that simultaneously work on arcs and assignments to combine the advantages of both approaches.

7.1. Initial solution computation

The proposed LNS starts from an initial feasible solution computed as follows: A relaxation of the original problem, which is obtained by dropping integrity constraints for variables X_{ij}, Y_k and Z_i, is solved to optimality. The relaxation is based on the substitution of constraints (9), (10) and (11) with constraints (22), (23) and (24), respectively.

$$0 \le X_{ij} \le 1 \qquad \forall i \in I \;\; \forall j \in I \tag{22}$$

$$0 \le Y_k \le 1 \qquad \forall k \in K \tag{23}$$

$$0 \le Z_i \le 1 \qquad \forall i \in I \tag{24}$$

To generate the initial solution, all variables Y_k with value 1 in the optimal solution of the relaxed problem are forced to be equal to 1 in the original problem as well. All the customers i assigned to ODs in the relaxed problem, i.e., $Z_i = 0$, are forced to be assigned to ODs in the original problem as well. Finally, all customers for whom $0 < Z_i < 1$ are neither forced to be assigned to ODs nor to be served by the owned fleet. We then solve the problem with these fixed variables for short execution time limit (T_{lim}^{init}) and keep the current best solution as the initial solution.

7.2. LNS Framework

Since the mathematical model proposed in Section 4 is able to solve only small-sized instances within reasonable run times, we have developed an LNS that solves the combined problem of selecting bid route customers if they are served by the company's fleet.

In this LNS, the algorithm starts from an initial feasible solution (see Section 7.1). A set of destroy operators is generated (see Section 7.3). Each of them follows a different rule to find a subset of customers to be removed from the routes of the owned vehicles. Furthermore, each operator identifies a promising subset of customers currently served by ODs as candidates for being served by owned vehicles.

The following rules are applied to restrict the set of possible moves allowed in the rebuilding phase:

- A customer cannot be inserted between two nodes if the arc between these nodes has not been removed by the destroy operator.
- Customers removed from the owned vehicles' routes can be assigned to any route or OD.
- Customers assigned to ODs in the current solution and candidates to be assigned to owned vehicles can be inserted into one of the owned vehicles' routes or assigned to any OD.
- Customers assigned to ODs in the current solution but not selected as candidates to be assigned to owned vehicles must be assigned to an OD.

These rules limit the number of solutions included in the neighborhood to explore but reduce the computational effort. Hence, we aim for a good balance between efficiency and effectiveness.

In each iteration, one of the destroy operators is randomly selected. Starting from the partially destroyed solution, a feasible solution is generated based on the previously described rebuilding rules. The model is run with a short time limit (T_{lim}), and the current best solution is selected. If the newly obtained solution is better than the best known solution, it is retained as the new best known solution. The algorithm terminates once the maximum number of iterations ($\mathrm{Iter}_{\max}$) is reached.

The additional constraints to be added to the model belong to two categories: (i) arc-fixing constraints, where for each *arc (i,j)* belonging to the current solution that has not been destroyed by the perturbation, we impose $X_{ij} = 1$, (ii) assignment-fixing constraints, where we force customers who were previously assigned to ODs and are not candidates to be inserted into the owned fleet's routes to be assigned to ODs. This can be obtained by imposing $Z_i = 0$ for all of them. All the remaining variables are eligible to be changed during the rebuilding phase.

7.3. Destroy operators

A set of destroy operators comprises the following five operators:

- **Random removal (RR)**: A set (P) of customers is randomly picked from among the customers currently assigned to the owned fleet. These customers are temporarily removed from the routing plan (by deleting their entering and exiting arcs). All customers currently assigned to ODs and within a radius of ρ from one of the customers in P are marked as candidates to be inserted into the owned fleet's routes.

- **Randomized worst removal (RWR)**: A set (P) of customers is drawn based on a probability that depends on the savings made from removing them from the route. Let pre_i and $succ_i$ be the predecessor and the successor, respectively, of i in the routing sequence. The savings associated with i can be computed as

$$sav_i = c_{pre_i i} + c_{i succ_i} - c_{pre_i succ_i}. \tag{25}$$

  All customers currently assigned to ODs and within a radius of ρ from one of the customers in P are marked as candidates to be inserted into the owned fleet's routes.
- **Clustered removal from routes (CR-R)**: One customer i is randomly chosen from among those served by the owned fleet in the current solution. Set P comprises i and all the other customers served by the owned fleet within a radius of ρ from i. All customers belonging to P are removed from the routes, and all customers currently assigned to ODs within a radius of ρ from one of the customers in P are marked as candidates to be inserted into the owned fleet's routes.
- **Clustered removal from ODs (CR-O)**: One customer i is randomly chosen from among those served by ODs in the current solution. Set P comprises i and all the other customers who are served by ODs and lie within a radius of ρ from i. All customers who are served by the owned fleet and are within a radius of ρ from one of the customers in P are removed from the routes. The operator's behaviour is similar to that of operator CR-R, but its scope is different. CR-R aims at removing a clustered set of customers from the owned fleet's routes, letting the exact model the decision to assign them to ODs. In order to favour such changes in the assignment, also customers, located nearby to the cluster, but currently assigned to ODs, are involved in the perturbation. The scope of CR-O, instead, is to attempt to insert in the routes a clustered set of customers currently assigned to ODs (not necessarily assigned to the same OD). To allow that, in the perturbation, we relax the assignment of a cluster of customers to ODs, by removing from the solutions all accepted bids involving at least one customer belonging to the cluster. In order to favour their insertion into the routes, all customers assigned to the owned fleet and located nearby to the customers in the cluster, are involved in the perturbation and temporarily removed from their route.
- **Bundle removal (BR)**: A bid k is randomly chosen from among those accepted in the current solution. Then all customers associated with bid k (τ_k) are considered candidates to be inserted into the owned fleet's routes. All customers served by the owned fleet and within a radius of ρ from one of the customers in τ_k are removed from the routes. Furthermore, in this paper, an additional condition is imposed for other operators: *all* customers in τ_k are assigned to (i) ODs or (ii) owned vehicles. This condition is imposed through a new set of constraints:

$$Z_i = Z_j \qquad \forall i \in \tau_k, j \in \tau_k : i \neq j \tag{26}$$

8. Computational experiments

In this section, we report the results obtained through an extensive computational study, which comprises three parts: (i) comparison of the performances of the two bundle-generation strategies, (ii) comparison of the performance of the LNS with respect to the MIP model solved by a commercial solver, and (iii) analysis of how ODs' characteristics (i.e., flexibility and willingness) impact the solution procedure.

To carry out these analyses, two sets of instances have been generated. The first comprises 20 small-sized instances, with 20 customers, 10 ODs, and 5 owned vehicles. In the first 10 instances, the customers' locations are distributed according to a random distribution in the square $[-5, 5]$, while in the last 10 instances, a clustered distribution has been adopted. For this, we randomly generate three seed customers β (where $\beta \in \{1, 2, 3\}$) with coordinates $[x_\beta, y_\beta]$ in the square $[-5, 5]$. For each of the first two seed customers, we generate six customers b (where $b \in \{1, .., 6\}$) with coordinates x_b randomly distributed in $[-1 + x_\beta, 1 + x_\beta]$ and y_b randomly distributed in $[-1 + y_\beta, 1 + y_\beta]$. For the third seed, we generate only five customers with the same procedure as for the other seeds; this is for the purpose of having 20 customers in total. We refer to these instances as small random (SR) and small clustered (SC), respectively. As mentioned before, ODs are characterized by flexibility and willingness, where flexibility indicates the accepted deviation from the shortest path, and willingness is reflected in an OD's bidding behavior. For both categories, we consider three levels: high, medium, and low. The maximum detour allowed for high, medium, and low flexibility is equal to 3,2, and 1, respectively. The willingness-associated multiplier is equal to 0.8, 1, and 2 for high, medium, and low levels, respectively.

Table 2
Impact of valid inequalities. The values of the objective function (OF), lower bounds (LB), and run times (TIME) are reported for both scenarios (valid inequalities included – VIs; valid inequalities excluded – NO VIs).

	VIs			NO VIs		
Instance	OF	LB	TIME	OF	LB	TIME
r-20-10-1	48.65	48.65	2.67	48.65	25.95	3600
r-20-10-2	48.17	48.17	20.53	48.17	19.71	3600
r-20-10-3	52.53	52.53	10.41	54.27	22.10	3600
r-20-10-4	53.04	53.04	24.62	56.51	25.51	3600
r-20-10-5	49.10	49.10	35.75	50.65	25.10	3600
r-20-10-6	50.13	50.13	40.81	51.17	25.73	3600
r-20-10-7	49.15	49.15	22.43	49.73	22.88	3600
r-20-10-8	50.06	50.06	1.986	53.50	25.56	3600
r-20-10-9	51.39	51.39	11.67	52.90	25.11	3600
r-20-10-10	53.96	53.96	12.10	54.04	24.21	3600
Avg	50.62	50.62	18.30	51.96	24.28	3600
Gap		0.00%			53.28%	

In our computations, we consider all nine combinations of flexibility and willingness, resulting in a total of 180 small-sized instances.

The second set comprises 10 large-sized instances, with 40 customers, 20 ODs, and 5 owned vehicles. The customers' locations are randomly distributed. We refer to these instances as *L*. From each instance, nine instances are derived considering different combinations of flexibility and willingness levels. This adds up to a total of 90 instances.

Computational tests have been executed on a system equipped with an Intel-i7-5500U processor running at 2.4 GHz clock speed and with 16 GB of RAM. All the procedures have been developed in the Xpress Mosel language. The MIP models – the one used in the bundle-generation phase and the one used to solve the VRP-OD-OB – are solved by the commercial solver Xpress 7.9. We set T_{lim} to 10 s, T_{lim}^{init} to 5 s, and $\text{Iter}_{\max}$ to 20 iterations, respectively. Set P comprises five customers. The maximum run time for solving the MIP is 3600 s.

The complete set of instances is publicly available [9].

8.1. Impact of valid inequalities

To show the benefit that can be obtained by adding the proposed valid inequalities (see Section 4), a comparison with and without valid inequalities has been conducted, with a time limit of 3600 s. These experiments have been performed on the small-sized instances with random customer distribution. The computational results are reported in Table 2. For each instance, the value of the objective function of the best solution (OF), the best lower bound (LB), and the computational time required (TIME) are re-

Table 3
Comparison of cluster-based (Clus) and corridor-based (Corr) bundling approaches on small-sized instances for different combinations of flexibility (Flex) and willingness (Will). We report the average values of total cost (Cost), run time (Time) in seconds, number of bundles (#Bundles), number of bids (#Bids), number of ODs (#OD), number of customers served by ODs (#Cust), and ratio of customers per OD (Cust/OD).

			Cost		Time		#Bundles		#Bids		#OD		#Cust		Cust/OD	
Instance	Flex	Will	Clus	Corr	Clus	Corr	Clus	Corr	Clus	Corr	Clus	Corr	Clus	Corr	Clus	Corr
SR	LOW	LOW	54.64	54.52	1681.85	641.64	47	18	98	33	2.1	1.5	2.7	4.6	2.23	1.45
SR	LOW	MED	52.81	50.88	550.90	20.06	47	18	98	33	2.9	2.8	6.4	9.5	2.33	3.45
SR	LOW	HIGH	47.56	45.29	103.41	3.518	47	18	98	33	5.2	4.1	11.7	13.5	2.37	3.38
SR	MED	LOW	54.64	54.51	1686.10	405.77	47	18	153	51	2.2	1.6	3.3	5.1	2.4	2.73
SR	MED	MED	52.27	50.62	577.02	18.30	47	18	153	51	3.3	2.9	7.9	10	2.84	3.49
SR	MED	HIGH	46.57	44.51	109.5	4.96	47	18	153	51	5.1	4	12.8	13.7	2.9	3.5
SR	HIGH	LOW	54.64	54.41	1621.07	526.89	47	18	194	66	2.3	1.7	3.2	5.5	1.4	2.23
SR	HIGH	MED	52.35	50.53	594.23	23.41	47	18	194	66	3.2	2.9	7.2	9.9	2.96	3.46
SR	HIGH	HIGH	46.38	44.43	114.27	6.95	47	18	194	66	5.1	4.1	13.1	14	2.98	3.48
Avg			51.32	49.97	782.04	183.50	47.00	18.00	148.33	50.00	3.49	2.84	7.59	9.53	2.49	3.02
% Impr				−2.63%		−76.54%		−61.70%		−66.29%		−18.47%		25.62%		−21.24%
SC	LOW	LOW	54.15	53.70	2506.80	1097.00	41	20	78	42	1.5	1.8	4.5	5.3	2.03	2.95
SC	LOW	MED	51.25	50.84	1316.10	704.70	41	20	78	42	2.0	2.4	6.3	7.5	3.24	3.20
SC	LOW	HIGH	47.38	46.53	580.60	373.7	41	20	78	42	2.8	3.1	9.8	9.5	3.53	3.2
SC	MED	LOW	54.64	53.11	1240.50	819.1	41	20	117	61	1.6	2	4.8	6.1	2.4	3.05
SC	MED	MED	50.34	49.75	660.8	483.90	41	20	117	61	2.2	2.6	7.1	8.1	3.3	3.14
SC	MED	HIGH	45.23	44.52	470.8	370.8	41	20	117	61	3.2	3.8	12.3	11.8	3.9	3.14
SC	HIGH	LOW	54.64	53.03	1284.5	695.6	41	20	153	76	1.8	2	5.5	6.3	2.92	3.3
SC	HIGH	MED	49.77	49.44	469.4	190.9	41	20	153	76	2.4	2.7	8	8.5	3.5	3.24
SC	HIGH	HIGH	44.27	43.69	126.1	8.7	41	20	153	76	3.5	3.9	13.2	12.8	3.8	3.34
Avg			50.19	49.40	961.73	527.16	41.00	20.00	116.00	59.67	2.33	2.70	7.94	8.43	3.18	3.17
% Impr				−1.56%		−45.19%		−51.22%		−48.56%		15.71%		6.15%		0.21%

Table 4
Comparison of cluster-based (Clus) and corridor-based (Corr) bundling approaches on large-sized instances for different combinations of flexibility (Flex) and willingness (Will). We report average values of total cost (Cost), run time (Time) in seconds, number of bundles (#Bundles), number of bids (#Bids), number of ODs (#OD), number of customers served by ODs (#Cust), and ratio of customers per OD (Cust/OD).

			Cost		Time		#Bundles		#Bids		#OD		#Cust		Cust/OD	
Instance	Flex	Will	Clus	Corr	Clus	Corr	Clus	Corr	Clus	Corr	Clus	Corr	Clus	Corr	Clus	Corr
L	LOW	LOW	109.23	103.67	4611.69	3611.99	222	41	540	128	7.6	4.2	14.9	15.1	2.45	3.62
L	LOW	MED	100.76	96.52	4634.28	3610.53	222	41	540	128	7.1	4.8	15.3	17.7	2.54	3.70
L	LOW	HIGH	86.44	83.72	3978.38	3611.75	222	41	540	128	8.2	6.9	18.3	25.5	2.37	3.71
L	MED	LOW	111.22	103.04	5286.13	3626.2	222	41	959	209	8	4.3	15	14.8	2.54	3.47
L	MED	MED	100.63	95.41	5309.25	3624.50	222	41	959	209	7.4	5.3	14.4	19.3	2.34	3.65
L	MED	HIGH	90.63	82.58	4887.86	1586.4	222	41	959	209	10.1	8	20.2	29.7	2.28	3.74
L	HIGH	LOW	109.69	102.75	6200.7	3641.1	222	41	1391	283	8.1	4.1	14.9	14.8	2.35	3.63
L	HIGH	MED	101.3	95.13	6240.07	3641.4	222	41	1391	283	7.4	5.1	14.9	18.9	2.41	3.72
L	HIGH	HIGH	90.15	81.4	5745.33	1005.3	222	41	1391	283	10.3	9	19.6	33.6	2.31	3.76
Avg			100.01	93.80	5210.41	3106.57	222.00	41.00	963.33	206.67	8.24	5.74	16.39	21.04	2.40	3.67
% Impr				−6.20%		−40.38%		−81.53%		−78.55%		−30.32%		28.41%		−52.85%

ported. We differ between the case where valid inequalities are added to the model (VIs) and the case without valid inequalities (NO VIs). If run times of 3600 are reported, the instances could not be solved to optimality within the given time limit. The last two rows report average values and the optimality gap ($(OF - LB)/LB$), respectively. As can be evinced from the reported results, the proposed valid inequalities are very effective and provide a strong reduction of run times. Therefore, all valid inequalities are included for the further experiments.

8.2. Comparison of bundle-generation strategies

To compare the two bundle-generation strategies, we can apply them to both small (SR and SC) and large-sized instances (L). Tables 3 and 4 show the results for the small and large-sized instances, respectively.

All three tables are organized as follows:

For each bundle-generation strategy, we report the following information: (i) best known objective function value (if the instance can be solved to optimality within the time limit), (ii) number of generated bundles, (iii) number of bids, (iv) number of ODs used, (v) number of customers served by ODs, and (vi) average number of customers per OD. In each strategy, we show the average values over 10 instances for each combination of flexibility and willingness. For each group of instances (*SR*, *SC*, and *L*), the average values and percentage improvements obtained by the corridor-based approach are reported. The computational times include the generation of bundles, generation of bids, and time for solving the MIP.

The results show that the proposed corridor-based approach outperforms the classical clustering-based approach. For the randomly distributed instances (SR), the total cost is, on average, 2.7% lower if the corridor-based approach is applied, and the average run time can be reduced by 86.7%. This is because the corridor-based approach generates fewer but more attractive bundles. Despite the fewer bundles generated and the lower average number of ODs engaged, the number of customers served by ODs is larger for the corridor-based approach (9.5 as compared to 7.6). This also indicates that the corridor-based approach can generate more attractive and profitable bundles for the ODs.

The same observations hold true for the clustered instances (SC) even though this customer distribution is more suitable for a clustering-based approach. Again, the corridor method obtains better results in both the total cost and run time. The number of bundles is higher than in the random case but still more than 50% lower as compared to the cluster approach. Finally, we can conclude that customers' geographical distribution does not affect the

Table 5
Comparison of MIP and LNS for large-sized instances. For the MIP, we report the best result obtained within a maximum run time of 3600 s. Column *Time* gives the required run time. For the LNS, we report the average gap to the MIP solution of (i) the best solution over 10 runs: Δ(best), (ii) the average solution over 10 runs: Δ(avg.), (iii) the average time when the best solution was found: *Found*, and (iv) the average total run time: *Time*.

		MIP		LNS			
Flex	Will	Cost	Time	Δ(best)	Δ(avg.)	Found	Time
Low	Low	103.7	3612	0.5%	1.4%	32.9	85.5
Low	Med	96.5	3611	0.2%	1.0%	36.1	96.5
Low	High	83.7	3612	0.5%	1.2%	23.2	93.8
Med	Low	103.0	3626	0.4%	1.4%	36.8	100.2
Med	Med	95.4	3625	0.3%	1.1%	26.2	98.6
Med	High	82.6	1586	2.0%	2.1%	14.6	87.1
High	Low	102.8	3641	0.5%	1.2%	34.7	105.3
High	Med	95.1	3641	0.3%	1.3%	37.0	91.9
High	High	81.4	1005	1.9%	2.1%	16.0	76.2
Avg.		93.8	3107	0.7%	1.4%	28.6	92.8

performance of the algorithms. Furthermore, the corridor-based approach performs better regardless of the distribution. Therefore, for large-sized instances, we have considered only randomly distributed customers.

The experiments conducted on large-sized instances confirm the dominance of the proposed corridor approach. Again, we can show that this approach reduces the total cost (−6.2%) and clearly outperforms the clustering-based approach in terms of computational time (−40.38%). The average optimality gap is 3.38% for the corridors-based approach and 9.19% for the clustering-based one.

We can also observe that the corridor-based approach generates fewer bundles but that the number of customers assigned to one bundle (i.e., OD) is higher. This shows that the corridor-based approach generates more attractive bundles. A comparison of Tables 3 and 4 indicates that these effects grow with the size of the instances. For what concerns the bundle generation phase, the corridor-based approach is relatively fast (less than 60 s), while the clustering method can take up to 1 h to generate bundles. Hence, the difference in computational times is not only due to the solving phase but mostly to the generation phase. Indeed, the corridors-based approach not only generates better bundles but also takes a much smaller time to generate them. This allows us to derive an important insight: a company should focus on providing fewer but more promising bundles. Concluding this analysis, we can state that the innovative corridor-based bundle generation approach is more suitable and profitable for the presented problem.

8.3. LNS Performance evaluation

As reported in Table 4, the computational times required to solve large-sized instances are relatively long, and only a few instances can be solved to optimality. Therefore, a more efficient approach is needed. We propose applying an LNS method, as described in Section 7, to obtain near-optimal solutions in a reasonable amount of time.

In Table 5, we compare the performances of the proposed LNS and the MIP solved by a commercial solver.

The comparison reveals that the LNS achieves good solutions (0.7% from the best solution found by the MIP) within considerably shorter computational times. On average, the run time of the LNS is more than 90% shorter than the time required by the MIP model. Moreover, the average results over 10 runs are close to the best results, which indicates that the LNS performs robustly.

In Table 6 we report a comparison between the best solutions obtained by MIP and by LNS within a time limit of 300 s. We report average results for each combination of flexibility and willingness. The average cost and time within the best solution has been found is reported. We observe that LNS achieves an improvement, on the solution quality, of 0.02% with respect to the MIP, and a saving of 87% in computational time. We further observe that MIP obtains slightly better solutions when the willingness to work is high, while LNS obtains better solutions on instances with low and medium willingness. This behavior can be explained by the fact that instances with high willingness are those in which, due to the lower bids offered by the drivers, the number of customers served by ODs is higher (see Table 4). Consequently, the routing plan is less complex, since it involves a small of number of customers. Indeed, the large computational times required by MIP are essentially due to the routing-related part of the model. Moreover, most of the solutions provided by MIP within 300 s on instances with high willingness, are optimal. Therefore, they cannot be improved by LNS. Thus, we can conclude that LNS, when compared against MIP, shows a good performance on all types of instances, requiring much smaller computational times.

Table 6
Comparison of best solutions obtained by MIP and by LNS within 300 s. Cost and time to reach the best solution (T.F.) are reported.

		MIP		LNS	
Flex	Will	Cost	T.F.	Δ(avg.)	T.F.
Low	Low	105.66	207.46	−0.55%	32.94
Low	Med	97.77	287.50	−0.28%	36.06
Low	High	86.80	197.40	0.72%	23.21
Med	Low	105.04	218.89	−0.50%	36.82
Med	Med	96.73	238.32	−0.32%	26.27
Med	High	83.75	200.42	0.68%	14.63
High	Low	104.55	223.54	−0.50%	34.71
High	Med	96.52	232.19	−0.13%	37.05
High	High	82.47	235.15	0.72%	15.96
Avg.		95.48	226.76	−0.02%	28.63

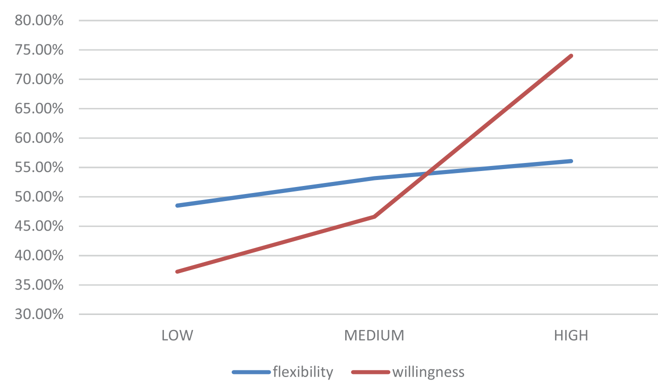

Fig. 3. Percentage of customers served by ODs for different levels of flexibility and willingness to work for large-sized instances.

8.4. Impact of ODs' flexibility and willingness to work

Further experiments have been conducted to assess the impact of the flexibility and willingness of ODs on the solution structure. For this purpose, we have compared the average ratios of customers served by ODs for each combination of flexibility and willingness. The results are displayed in Fig. 3. We can see that the ratio is more sensitive to the OD's willingness than their flexibility. In fact, for a high flexibility level, 56.08% of customers are assigned to ODs, while the percentage decreases to 48.5% if the flexibility in deviating from the shortest path is low.

The impact of willingness is stronger. If the ODs are characterized by a high willingness to work, i.e.,if they put forth competitive efforts, 74% of customers are served by ODs, while this percentage

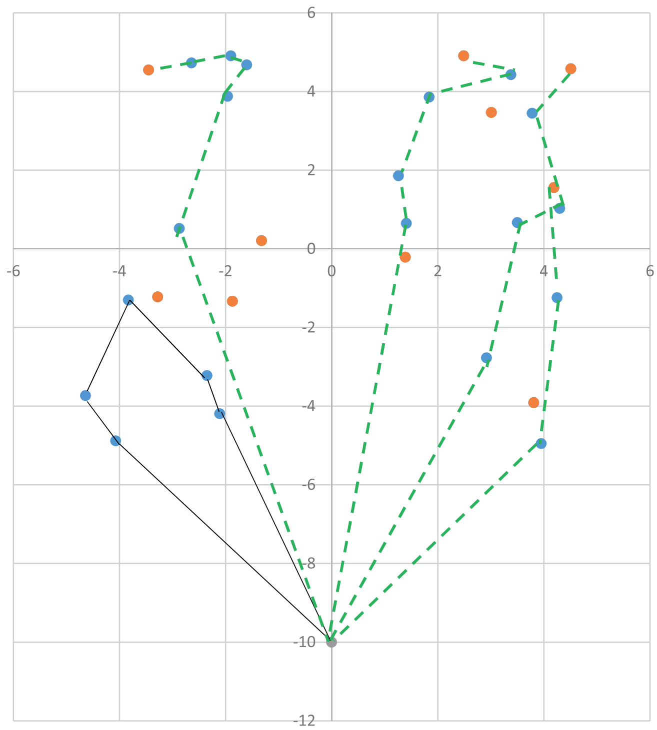

Fig. 4. Optimal solution for instance 20 – 10 – 1 with high flexibility and high willingness. Customers are depicted as light blue circles and ODs as orange circles. Bold lines represent the routes covered by the owned fleet, while dotted lines represent the ODs' paths. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Table 7
Comparison of cluster-based (Clus) and corridor-based (Corr) bundling approaches on instances with clustered OD distribution.

Instance	Clus	Corr
clustOD1	45.51	41.82
clustOD2	45.44	40.22
clustOD3	52.13	49.53
clustOD4	45.37	47.57
clustOD5	47.33	44.60
clustOD6	48.80	47.39
clustOD7	48.37	47.65
clustOD8	60.10	59.76
clustOD9	51.74	52.27
clustOD10	49.36	51.97
Avg	49.41	48.28
% Impr		2.30%

decreases to 37.25% if their willingness is low. In general, we can observe that if ODs show medium flexibility and medium willingness, the percentage of customers assigned to them is about 48%. However, for the most favorable combination, i.e., high flexibility and high willingness, an average of 84% of customers are served by ODs. This indicates that the optimal solution for the system lies in a mixed-distribution strategy that integrates the owned fleet and ODs. An example for such a solution for one of the small-sized instances is displayed in Fig. 4.

8.5. Impact of clustered OD distribution

In this subsection, we describe the experiments carried out on small-sized instances with clustered OD distribution in order to compare the performance of the proposed bundle-generation approaches. The layout of one of these instances, namely *clustOD1*, is reported in Fig. 5. The results are in Table 7, which shows the

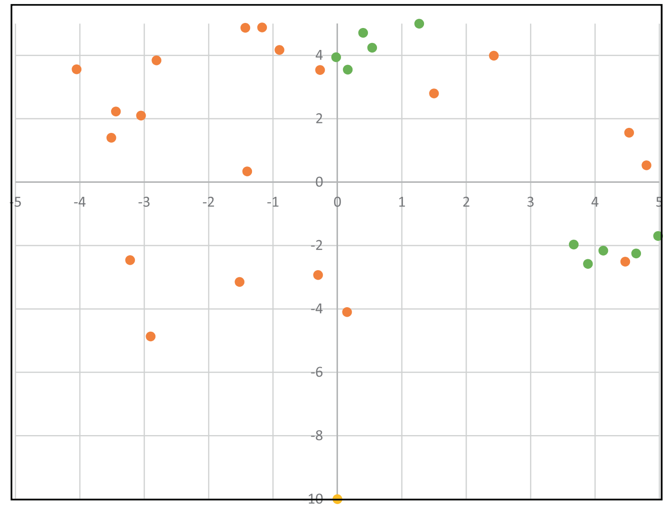

Fig. 5. Layout of instance clustOD1.

optimal values for both bundling approaches. The last two rows show the average values and percentage improvements obtained by the corridor-based approach as compared to the cluster-based approach. We can see that the corridor-based approach performs better in 70% of the cases. It provides an average improvement of 2.30%, which supports the results obtained for the randomly distributed ODs. Thus, we can conclude that the comparative performances of the two approaches are not influenced by the ODs' distribution.

8.6. Dynamic appearance of ODs

In many real-life applications, ODs are customers who have performed an online purchase and plan to collect their order from the store. This is a well-known method, "click&collect," which is offered by several retail chains. In such a context, information about a potential OD's availability is known at least one day in advance. Hence, the problem can be described by a static model. Nevertheless, there are other contexts in which such information is not available a priori. Instead, an OD's availability is learned dynamically during the day. For this setting, we have assumed that the customers' demands are known in advance, while the available ODs are revealed at fixed time intervals. Thus, first, the static problem is solved considering virtual ODs with medium flexibility and willingness. Furthermore, we have assumed that the ODs' locations are equally distributed over the customers' areas (see Fig. 6).

Orders that were assigned to the owned fleet in the optimal (or best) solution are loaded into the owned fleet vehicles, and these vehicles start their daily routing plan. Such decisions are kept fixed and cannot be modified during the day. However, all the orders that were assigned to the virtual ODs are kept in the store, waiting for potentially interested ODs. At each time interval, the situation is re-evaluated since new ODs may appear dynamically. If this happens, the model is solved again, considering only unserved orders. If one or more bids are accepted, the corresponding orders are assigned to ODs, while the remaining orders are kept in the store waiting for more ODs. At the last time interval, the orders that have not been assigned to ODs are served with the owned fleet, for which a standard VRP is solved. The total cost is computed as the sum of the owned-fleet routing costs for the delivery plan that started at the beginning of the day, the costs associated with the accepted bids, see (1), and the cost of the supplementary routing plan for the owned fleet to perform the delivery of the remaining orders. The goal of these experiments is to evalu-

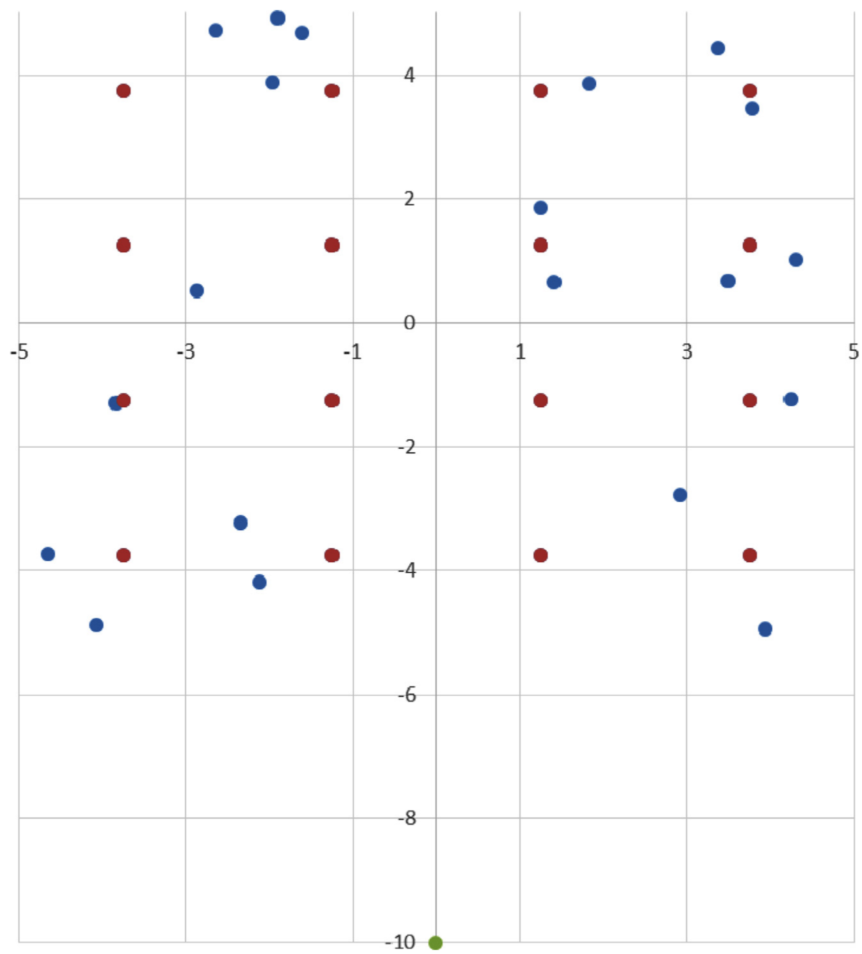

Fig. 6. Instance r20-10-1 with equispaced virtual ODs.

Table 8
Results of the small-sized instances with randomly distributed customers and dynamic arrivals of ODs.

	Clus		Corr	
Instance	Expected Cost	Real Cost	Expected Cost	Real Cost
r-20-10-1	54.54	54.54	53.22	51.97
r-20-10-2	54.04	53.12	50.34	50.15
r-20-10-3	54.77	57.07	48.46	44.71
r-20-10-4	51.58	52.89	54.02	54.47
r-20-10-5	51.38	53.21	49.65	46.81
r-20-10-6	54.25	55.08	50.92	52.20
r-20-10-7	51.12	55.65	49.96	47.88
r-20-10-8	57.98	57.70	57.64	57.16
r-20-10-9	54.12	55.28	55.45	53.75
r-20-10-10	56.19	55.79	57.64	56.88
Avg	53.00	55.03	52.73	51.60
% Impr			2.34%	6.24%

ate the proposed corridor-based approach in a dynamic context. The experiments have been performed on a set of small-sized instances with randomly generated customer locations. We have assumed 16 virtual ODs, which are considered to compute an initial plan. These are equispaced across the customers' area, as shown in Fig. 6. We have considered four re-evaluation time intervals. In each of them, an equally distributed random number between 0 and 10 gives the amount of new ODs. Each of them is associated with a random location, flexibility, and willingness. Each instance is solved five times with different OD-appearance scenarios. The results are shown in Table 8. For each instance, we report the average (over five runs) expected and real costs for both the cluster-based and corridor-based bundling approaches. The penultimate row reports average results over all instances, while the last one reports the percentage of improvement obtained with the corridor-based approach on the expected and total costs. It is noteworthy that the expected costs are the value of the optimal solution of the static problem in which the virtual ODs are used. As shown in Table 8, the corridor-based approach outperforms the cluster-based one. The expected cost is, on average, 2.34% lower, which supports the results observed for the pure static problem (see 8.2). The results show that the corridor-based approach not only allows better a priori plans but also yields better matching with dynamically appearing ODs. This is reflected in the reduction of the real costs by 6.24% on average. Thus, we can conclude that the advantage of the proposed corridor-based approach becomes even clearer in the considered setting of dynamically arriving ODs.

8.7. Real road network

The bundles generated by the corridor-based approach may involve customers who are far from each other but close to the corridor of an OD (see Fig. 2). On the contrary, the bundles generated by the cluster-based approach are typically more compact. The previous experiments performed in different settings, both static and dynamic, have all shown a dominance of the corridor-based approach. However, all these experiments have been carried out based on Euclidean distances among the nodes, which does not reflect the real-world setting; for instance, ODs may use arterial roads, and thus, small distances may have a strong impact since

Table 9
Results obtained on instances generated on a real road network. We report optimal cost obtained with cluster-based (Clus) and corridor-based (Corr) bunding.

Instance	Clus	Corr
RRN1	43.67	43.12
RRN2	33.46	32.44
RRN3	32.40	31.35
RRN4	31.27	30.37
RRN5	26.10	23.76
RRN6	29.92	31.23
RRN7	26.83	25.93
RRN8	25.50	24.60
RRN9	25.34	24.44
RRN10	23.63	22.71
Avg	29.81	28.99
Impr %		2.82%

leaving the road may be disproportionately more time consuming than reflected by Euclidean distances. To capture these effects on the proposed corridor-based approach, we have performed experiments on a real road network. A set of 10 instances with 10 customers and 3 ODs have been generated on the real road network of Milan. The results are shown in Table 9. For each instance, the optimal cost obtained with each bundling approach is reported. The last two rows report the average values and percentage improvements obtained by the corridor-based approach. As seen in Table 9, the corridor-based approach works well on the real road network. It obtains better results in 9 out of 10 instances, with an average percentage improvement of 2.82%, which further supports the results of the artificially generated instances.

9. Conclusions and future directions

In this paper, we introduced the vehicle routing problem with occasional drivers (ODs) and bundles of orders (VRP-OD-OB). A mathematical formulation and valid inequalities were proposed. This work contributes to the existing literature by considering the possibility of assigning bundles of customers, rather than single orders, to drivers. Furthermore, we proposed a bidding system, where drivers submit their bids for bundles they are willing to serve. These bids depend on the detour required to serve the customers included in the bundle and on the OD's level of willingness to perform the deliveries. The system's objective is to decide (i) which bundles should be assigned to which OD and (ii) which customers should be served by the fleet of company-owned vehicles. Since the number of feasible bundles grows exponentially with the size of the problem, only a subset of bundles can be offered for bidding. To solve this, we proposed an innovative bundle-generation technique based on geographical corridors. We compared this approach to a more traditional clustering method. Extensive computational experiments showed that the newly proposed corridor-based approach strongly outperforms the more traditional clustering-based one. The computational results revealed that the new approach creates attractive and profitable bundles within considerably shorter computational times. Additionally, the number of bundles generated by the corridor-based approach is lower; therefore, the overall problem can be solved in a shorter amount of time. We have compared the performances of the two bundling approaches in different problem settings: (i) when ODs' final destinations are clustered, (ii) when the availability of ODs is not known in advance but is revealed at fixed time intervals along the working day, and (iii) when distances are computed on a real road network. In all these three cases, the corridor-based approach has been shown to be more effective in generating attractive and profitable bundles, also yielding reduced total costs. In the case of dynamically appearing ODs, considerable total cost reductions were shown. This holds true for all of the considered settings. Thus, there is strong evidence that the proposed approach is of high relevance for practical applications. Moreover, from this analysis, we derived an important managerial insight. It is not necessary to provide a high number of bundles to achieve good quality solutions. Instead, companies should focus on generating fewer but more attractive bundles.

While small-sized instances can be solved to optimality, we proposed a large neighborhood search (LNS)-based matheuristic (MH) to solve larger instances. This method obtained near-optimal solutions (less than 1% from the best known solution) in a short computational time. Finally, we provided managerial insights on the impact of drivers' characteristics on the ratios of customers served by ODs.

All instances have been made publicly available [9] to encourage other researchers to contribute to this highly relevant and dynamically evolving field.

Further developments in this field could address potential savings that can be achieved by applying the corridor-based approach to generate appealing bundles for ODs operating in other contexts, such as multi-echelon distribution systems (in which they could be in charge of the last leg of the distribution), multi-echelon reverse logistics systems (in which they perform the first leg), or highly dynamic pickup-and-delivery systems (such as food delivery).

CRediT authorship contribution statement

Simona Mancini: Conceptualization, Investigation, Methodology, Software, Validation, Writing – original draft, Writing – review & editing. **Margaretha Gansterer:** Conceptualization, Funding acquisition, Methodology, Investigation, Validation, Supervision, Writing – original draft, Writing – review & editing.

Acknowledgements

This work is supported by FWF Austrian Science Fund: P 34502-N and P 34151-N.

References

[1] United Nations. Press release 007/2019.
[2] Desmichel P, Kocher B. Luxury single- versus multi-brand stores: the effect of consumers' hedonic goals on brand comparisons. J Retailing 2020;96(2):203–19.
[3] Wu X, Liao H. Modeling personalized cognition of customers in online shopping. Omega 2021;104:102471.
[4] He B, Gupta V, Mirchandani P. Online selling through o2o platform or on your own? Strategic implications for local brick-and-mortar stores. Omega 2021;103:102424.
[5] Huang S, Potter A, Eyers D, Li Q. The influence of online review adoption on the profitability of capacitated supply chains. Omega 2021;105:102501.
[6] Yildiz B, Savelsbergh M. Service and capacity planning in crowd-sourced delivery. Transp Res Part C EmergTechnol 2019;100:177–99.
[7] Ulmer M, Savelsbergh M. Workforce scheduling in the era of crowdsourced delivery. Transp Sci 2020;54(4):1113–33.
[8] Dai H, Liu P. Workforce planning for O2O delivery systems with crowdsourced drivers. Ann Oper Res 2020;291:219–45.
[9] Mancini S, Gansterer M. Vehicle routing with occasional drivers and order bundles (VRP-OD-OB) instances. Mendeley Data, V2 2020. doi:10.17632/rcbd675rt62.
[10] Archetti C, Bertazzi L. Recent challenges in routing and inventory routing: e–commerce and last-mile delivery. Networks 2021;77(2):255–68.
[11] Le T, Ukkusuri S. Influencing factors that determine the usage of the crowd–shipping services. Transp Res Rec JTransp Res Board 2019;2673(7):550–66.
[12] Ermagun A, Shamshiripour A, Stathopulos A. Performance analysis of crowd–shipping in urban and suburban areas. Transportation 2020;47:1955–85.
[13] Simoni M, Marcucci E, Gatta V, Claudel G. Potential last-mile impacts of crowdshipping services: a simulation-based evaluation. Transportation 2020;47:1933–54.
[14] Alnaggar A, Gzara F, Bookbinder JH. Crowdsourced delivery: a review of platforms and academic literature. Omega 2021;98:102139.
[15] Allahviranloo M, Baghestani A. A dynamic crowdshipping model and daily travel behavior. Transp Res Part E LogistTransp Rev 2019;128:175–90.

[16] Le TV, Ukkusuri SV, Xue J, Van Woensel T. Designing pricing and compensation schemes by integrating matching and routing models for crowd-shipping systems. Transp Res Part E LogistTransp Rev 2021;149:102209.
[17] Archetti C, Savelsbergh M, Speranza MG. The vehicle routing problem with occasional drivers. Eur J Oper Res 2016;254(2):472–80.
[18] Macrina G, Di Puglia Pugliese L, Guerriero F, Laganà D. The vehicle routing problem with occasional drivers and time windows. In: Sforza A, Sterle C, editors. Optimization and decision science: methodologies and applications. Cham: Springer International Publishing; 2017. p. 577–87. ISBN 978-3-319-67308-0
[19] Macrina G, Guerriero F. The green vehicle routing problem with occasional drivers. Cham: Springer International Publishing; 2018. p. 357–66.
[20] Macrina G, Di Puglia Pugliese L, Guerriero F. Crowd-shipping with time windows and transshipment nodes. Comput Oper Res 2020;113:104806.
[21] Arslan A, Agatz N, Kroon A, Zuidwijk R. Crowdsourced delivery: a dynamic pickup and delivery problem with ad-hoc drivers. Transp Sci 2019;53(1):222–35.
[22] Dayarian I, Savelsbergh M. Crowdshipping and same-day delivery: employing in-store customers to deliver online orders. Prod Oper Manage 2020;29(9):2153–74.
[23] Gatta V, Marcucci E, Nigro M, Patella SM, Serafini S. Public transport-based crowdshipping for sustainable city logistics: assessing economic and environmental impacts. Sustainability 2018;11(1):1–14.
[24] Gatta V, Marcucci E, Nigro M, Serafini S. Sustainable urban freight transport adopting public transport-based crowdshipping for B2C deliveries. Eur Transp Res Rev 2019;11:13.
[25] Lin X, Nishiki Y, Tavasszy L. Performance and intrusiveness of crowdshipping systems: an experiment with commuting cyclists in the netherlands. Sustainability 2020;12(7):1–14.
[26] Kafle N, Zou B, Lin J. Design and modeling of a crowdsource-enabled system for urban parcel relay and delivery. Transp Res Part B Methodol 2017;99:62–82.
[27] Behrend M, Meisel F. The integration of item-sharing and crowdshipping: can collaborative consumption be pushed by delivering through the crowd? Transp Res Part B Methodol 2018;111:227–43.
[28] Behrend M, Meisel F, Fagerholt K, K A. An exact solution method for the capacitated item-sharing and crowdshipping problem. Eur J Oper Res 2019;279(2):589–604.
[29] Dahle L, Andersson K, Christiansen M, Speranza MG. The pickup and delivery problem with time windows and occasional drivers. Comput Oper Res 2019;109:122–33.
[30] Boysen N, Emde S, Schwerdfeger S. Crowdshipping by employees of distribution centers: optimization approaches for matching supply and demand. Eur J Oper Res 2022;296(2):539–56.
[31] Horner H, Pazour J, Mitchell JE. Optimizing driver menus under stochastic selection behavior for ridesharing and crowdsourced delivery. Transp Res Part E LogistTransp Rev 2021;153:102419.
[32] Mofidi SS, Pazour JA. When is it beneficial to provide freelance suppliers with choice? A hierarchical approach for peer-to-peer logistics platforms. Transp Res Part B Methodol 2019;126:1–23.
[33] Gansterer M, Hartl RF. Centralized bundle generation in auction based collaborative transportation. OR Spectr 2018;40(3):613–35.
[34] Gansterer M, Hartl RF, Sörensen K. Pushing frontiers in auction-based transport collaborations. Omega 2019;94:102042.
[35] Englmaier F, Guillén P, Llorente L, Onderstal S, Sausgruber R. The chopstick auction: a study of the exposure problem in multi-unit auctions. Int J Ind Organiz 2009;27(2):286–91.
[36] Berger S, Bierwirth C. Solutions to the request reassignment problem in collaborative carrier networks. Transp Res Part E LogistTransp Rev 2010;46:627–38.
[37] Gansterer M, Hartl RF. Request evaluation strategies for carriers in auction-based collaborations. OR Spectr 2016;38(1):3–23.
[38] Pisinger D, Ropke S. Large neighborhood search. In: Gendreau M, Potvin J-Y, editors. Handbook of metaheuristics. Boston, MA: Springer US; 2010. p. 399–419. ISBN 978-1-4419-1665-5
[39] Mancini S. A real-life multi depot multi period vehicle routing problem with a heterogeneous fleet: formulation and adaptive large neighborhood search based matheuristic. Transp Res Part C EmergTechnol 2016;70:100–12.
[40] Mancini S. The hybrid vehicle routing problem. Transp Res Part C EmergTechnol 2017;78:1–12.
[41] Mancini S, Stecca G. A large neighborhood search based matheuristic for the tourist cruises itinerary planning. Comput Ind Eng 2018;122:140–8.