CIRRELT-2023-35

An Exact Method for a Last-mile Delivery Routing Problem with Multiple Deliverymen

Fernando Senna
Leandro C. Coelho
Reinaldo Morabito
Pedro Munari

October 2023

Document de travail également publié par la Faculté des sciences de l'administration de l'Université Laval, sous le numéro FSA-2023-008

Bureau de Montréal
Université de Montréal
C.P. 6128, succ. Centre-Ville
Montréal (Québec) H3C 3J7
Tél : 1-514-343-7575
Télécopie : 1-514-343-7121

Bureau de Québec
Université Laval,
2325, rue de la Terrasse
Pavillon Palasis-Prince, local 2415
Québec (Québec) G1V 0A6
Tél : 1-418-656-2073
Télécopie : 1-418-656-2624

An Exact Method for a Last-mile Delivery Routing Problem with Multiple Deliverymen

Fernando Senna[1], Leandro C. Coelho[2], Reinaldo Morabito[1], Pedro Munari[1]

1. Industrial Engineering Department, Federal University of São Carlos, Brazil
2. Interuniversity Research Centre on Enterprise Networks, Logistics and Transportation (CIRRELT), Canada research chair in integrated logistics, Université Laval, Canada

Abstract. The demand for efficient last-mile delivery systems in large cities creates an opportunity to develop innovative logistics schemes. In this paper, we study a problem in which each vehicle may travel with more than one deliveryman to serve multiple customers with a single stop of the vehicle, increasing the delivery efficiency. We extend the vehicle routing problem with time windows and multiple deliverymen by explicitly considering the deliveryman routes. We introduce the problem, formally define it with a novel formulation, propose valid inequalities, and develop a tailored branch-and-Benders-cut (BBC) algorithm to solve it. The BBC is capable of solving 89% of the instances to proven optimality in reasonable times, many of them of realistic sizes. Additionally, we show the benefits of evaluating the deliveryman routes considering a cost minimization perspective, and discuss relevant solutions for urban logistics problems that can help decrease congestion and emissions.

Keywords: routing, last-mile delivery, branch-and-cut, Benders decomposition, multiple deliverymen.

Acknowledgements. This work was supported by São Paulo Research Foundation (FAPESP) [grant numbers 2013/07375-0, 2021/14441-5, 2022/09679-5, 2022/05803-3], Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) [Finance Code 001], the National Council for Scientific and Technological Development (CNPq) [grant number 313220/2020-4, 405702/2021-3], and the Natural Sciences and Engineering Research Council of Canada (NSERC) [grant number 2019-00094]. This support is greatly acknowledged. We thank the Digital Research Alliance of Canada for providing high-performance parallel computing facilities. We also thank Prof. Antonio Carlos Moretti for his support in the early stages of development of this work.

---

* Corresponding author: leandro.coelho@fsa.ulaval.ca

Dépôt légal – Bibliothèque et Archives nationales du Québec
Bibliothèque et Archives Canada, 2023

1 Introduction

The increasing demand for cost- and time-efficient delivery in densely populated urban areas creates additional challenges for last-mile delivery systems, such as poor traffic conditions and difficulty in finding parking locations (Martinez-Sykora et al., 2020; Boysen, Fedtke, and Schwerdfeger, 2021). However, the proximity of customers allows for inventive developments to overcome these challenges. For instance, the combination of trucks and drones is already well-known (Li et al., 2021) since the seminal work by Murray and Chu (2015). Similarly, the combination of robots and trucks has also been applied to last-mile delivery systems (Alfandari, Ljubić, and De Melo da Silva, 2022). Alternatively, one could rely on crowd-sourcing operations in last-mile delivery, as proposed by Ouyang, Leung, and Huang (2023), or on combining vehicles, cargo bikes, and walking porters, such as in the problem presented by Bayliss et al. (2023).

Another well-adopted possibility in city logistics is the combination of vehicles with walking carriers (Wehbi, Bektaş, and Iris, 2022; Le Colleter et al., 2023). In particular, Pureza, Morabito, and Reimann (2012) proposed the vehicle routing problem with time windows and multiple deliverymen (VRPTWMD), which arose from a practical application of last-mile delivery from a beverage company. In this problem, a vehicle may travel with more than one deliveryman. Once the vehicle parks, the deliverymen walk to serve the customers in parallel. This reduces the time the vehicle stays parked throughout the route, allowing it to serve more customers in a single route. Therefore, a smaller fleet of vehicles can serve the same customers compared to the traditional approach of having a single deliveryman traveling in each vehicle. Since deliverymen fixed costs are smaller than those of the vehicles, this creates an opportunity for operational cost reduction.

The VRPTWMD is often modeled using a network given by nodes that correspond to clusters of customers (Pureza, Morabito, and Reimann, 2012; Álvarez and Munari, 2017; Munari and Morabito, 2018). Clusters are defined in advance, in a previous decision stage, and the service time at a cluster depends on the number of deliverymen in the vehicle that visits that cluster. Hence, at each stop of a vehicle at a node, the service time at this node is the service time of the cluster divided by the number of deliverymen on the vehicle. Some variants consider the definition of the clusters as an endogenous decision, thus determining also the clustering of customers that are visited at each stop of the vehicles, as in Senarclens de Grancy and Reimann (2015). However, in these variants, the authors still simply divide the service time of a cluster by the number of deliverymen that serve it. To the best of our knowledge, no study has addressed the VRPTWMD and related variants explicitly considering the routes traveled by the deliverymen inside the clusters. Moreover, authors have assumed thus far that the deliverymen capacities are small compared to the customer demands, such as in the beverage industry from which the problem emerged, making the deliveryman routes trivial. However, in applications where the customer demands are small (e.g., e-commerce) or the deliverymen capacities are large (e.g., deliverymen with small carts or cargo bikes), this assumption is not valid and the deliveryman routes can significantly affect the vehicle routes.

In this paper, we extend the VRPTWMD by also designing the deliveryman routes inside each cluster, instead of simply considering round-trips. Since most drone-truck and robot-truck combinations consider that drones and robots can only visit one customer at a time (Moshref-Javadi and Winkenbach, 2021; Ostermeier, Heimfarth, and Hübner, 2023), our work also generalizes such problems. Furthermore, to efficiently solve the problem, we propose a Benders decomposition-based exact algorithm (Benders, 1962), which might be of broader interest given that the majority of works that address the VRPTWMD and related problems rely on heuristics (Pureza, Morabito, and Reimann, 2012; Senarclens de Grancy and Reimann, 2014; Moshref-Javadi and Winkenbach, 2021; Wehbi, Bektaş, and Iris, 2022; Le Colleter et al., 2023).

The contributions of this paper are threefold. First, we introduce a novel problem in the literature with practical and theoretical relevance, namely the vehicle routing problem with time windows, multiple deliverymen, and two-level routing (VRPTWMD2R). Second, we present a formulation for this problem and introduce several families of valid inequalities that tighten the linear programming (LP) relaxation of this formulation. Third, we propose a branch-and-Benders-cut method to solve the problem, which is an exact algorithm based on Benders decomposition, and develop lower bounding techniques.

The remainder of this paper is organized as follows. Section 2 reviews the pertinent literature. In Section 3, the problem is defined. Section 4 introduces the mathematical formulation and valid inequalities. Section 5 describes the exact algorithm to solve the problem. In Section 6, the computational experiments are outlined and the results are evaluated. Finally, Section 7 presents concluding remarks.

2 Literature review

Pureza, Morabito, and Reimann (2012) introduced the VRPTWMD as a variant of the classical vehicle routing problem (VRP). In this variant, in addition to time windows and vehicle capacity constraints, the vehicles may carry more than one deliveryman to reduce overall service time. The problem arises from companies that make regular deliveries in densely populated urban areas, in which the proximity of customers creates the possibility of serving more than one customer with a single stop of the vehicle. In such case, the presence of multiple deliverymen allows the customers to be served in parallel, reducing the time of each stop of the vehicles. Since the vehicle fixed costs are usually higher than those of the deliverymen, increasing the number of deliverymen can reduce the number of vehicles needed, decreasing the overall costs.

The problem dynamics are based on the customers with similar time windows and close to each other being previously grouped in clusters. The vehicles travel from the depot to the clusters and, once they arrive, the deliverymen leave the vehicle to serve the customers. Once all customers in a cluster are served, the deliverymen return to the vehicle and travel to the next cluster on the vehicle route.

Several authors have studied this problem with different approaches. Pureza, Morabito, and Reimann (2012) compared the performance of two metaheuristics: tabu search (TS) and ant colony optimization (ACO). Senarclens de Grancy and Reimann (2014) systematically compared the performance of ACO and greedy randomized adaptative search procedure (GRASP) to solve the problem. Álvarez and Munari (2016) solved the problem with iterated local search (ILS) and large neighborhood search (LNS). Munari and Morabito (2018) proposed the first exact algorithm for the problem, which consisted of a branch-price-and-cut method, thus based on the column generation technique. Álvarez and Munari (2017) combined this exact method with the metaheuristics ILS and LNS, resulting in a hybrid method for the problem. Souza Neto and Pureza (2016) proposed a variant of the VRPTWMD in which vehicles can perform more than one route and solved it with GRASP, a commercial solver, and a hybrid method.

All of the above-mentioned studies address the problem considering two simplifying hypotheses: (i) the clusters are predefined, and (ii) the time spent in each cluster is approximated by a function of the cluster demand and the number of deliverymen, ignoring the routes traveled by the deliverymen. To incorporate clustering issues, Senarclens de Grancy and Reimann (2015) proposed two heuristics to cluster the customers, and Senarclens de Grancy (2015) combined these heuristics in an iterative method to optimize clustering and routing.

We are not aware of any study addressing the design of deliveryman routes within the VRPTWMD. Approximating the service time of clusters based on their demand and the number of deliverymen may be reasonable when the deliverymen capacities are small compared to the customers demands. In such cases, the deliverymen can only visit one customer in each of their routes, making the optimal deliveryman routes trivial (i.e., round trips), with no need to be optimized. However, when the deliverymen capacities are large compared to the customers demands, they can visit more than one customer in each route. In such cases, approximating the cluster service time based on the demand and the number of deliverymen becomes less accurate and does not represent the problem complexity. This assessment is important because it affects all of the other decisions of the problem, namely the number of vehicles and deliverymen, and the vehicle routes. The present study addresses this issue by generalizing the VRPTWMD to consider two-level routing (VRPTWMD2R), i.e., both the vehicle and the deliveryman routes.

3 Problem definition

We define the VRPTWMD2R considering different graph representations for vehicle and deliveryman routes (first- and second-level). For vehicle routes, we assume a single depot and a set of clusters $N = \{1, 2, \dots, n\}$, where $n > 0$ is the number of clusters. Each cluster consists of a set of customers and a single parking location. We shall refer to clusters and parking locations interchangeably. Let $G = (N_0, A)$ be a directed graph, where $N_0 = \{0, n+1\} \cup N$ is the set of nodes and $A = \{i, j \in N_0 \mid i \neq j,\ i \neq n+1,\ j \neq 0\}$

is the set of arcs. Indices 0 and $n+1$ represent the depot, and all vehicle routes start at 0 and end at $n+1$. This graph only concerns the nodes and arcs related to the design of first-level routes.

For each cluster $i \in N$, we define a directed graph $G^i = (N_0^i, A^i)$, given by the set of nodes $N_0^i = \{0_i, n_i + 1\} \cup N^i$, where N^i is the set of n_i customer nodes in this cluster and $A^i = \{h, k \in N_0^i \mid h \neq k,\ h \neq n_i + 1,\ k \neq 0_i\}$ is the set of arcs related to the second-level routes inside this cluster. Nodes 0_i and $n_i + 1$ represent the parking location, and the deliveryman routes must depart from 0_i and return to $n_i + 1$, traversing only the arcs in A^i. Both nodes 0_i and $n_i + 1$ are at the same place as the corresponding parking location $i \in N$. No customer is part of more than one cluster, i.e., $N^i \cap N^j = \emptyset, \forall\ i, j \in N, i \neq j$. To make the notation clear, we shall represent nodes of first-level routes (set N) by i and j, and those of second-level routes (sets $N^i, i \in N$) by h and k.

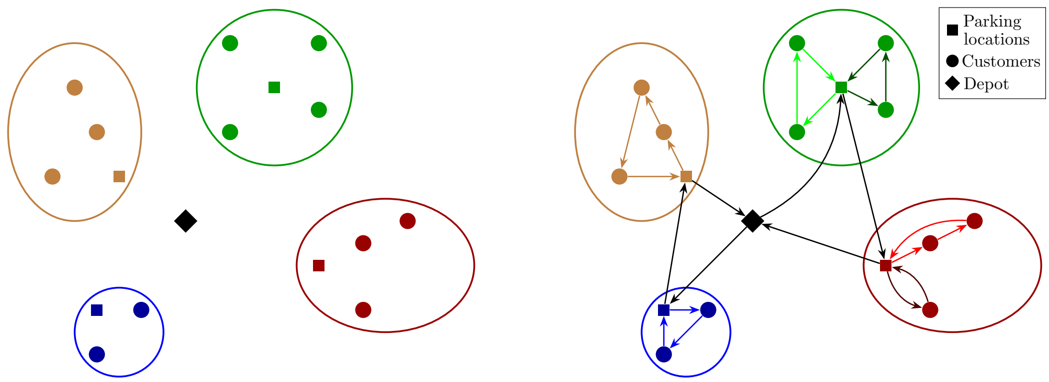
Every cluster is served by exactly one vehicle, and every customer inside a cluster is served by exactly one deliveryman. Both clusters and customers have time windows that indicate when the service may begin, which are supposed to be compatible in order to ensure feasibility. Customers have positive demands that are aggregated to define cluster demands, typically consisting of a couple of customers. We assume that deliverymen do not have capacity constraints since clusters are relatively small, and hence all customers of a cluster could be served by a single deliveryman when considering only capacity constraints.

Each vehicle may travel with up to M_L deliverymen. Once the vehicle arrives at a cluster, the deliverymen leave it to serve the customers. After serving all of them, the deliverymen return to the vehicle and it travels to the next cluster in the route. We define the set of possible numbers of deliverymen in a vehicle as $L = \{1, 2, \ldots, M_L\}$. We assume that the vehicle fleet and the deliveryman team are both homogeneous.

The decisions of the problem are (i) the number of vehicles to be used, (ii) the number of deliverymen in each vehicle, (iii) the vehicle routes, and (iv) the deliveryman routes. These decisions should be made ensuring that every customer is served, and respecting time windows and vehicle capacity. The goal of the problem is to minimize the fixed costs associated with vehicles and deliverymen and the distance-related costs of vehicle and deliveryman routes.

Figure 1 illustrates the VRPTWMD2R. Figure 1a presents an instance of the problem, with the customers clustered around their respective parking locations. Figure 1b represents a feasible solution to the problem. The black arrows that travel among parking locations represent vehicle paths, while the colored arrows inside the clusters show deliveryman routes. In the picture, the vehicle that serves the clusters on the left-hand side of the picture travels with one deliveryman and the other one travels with two deliverymen.

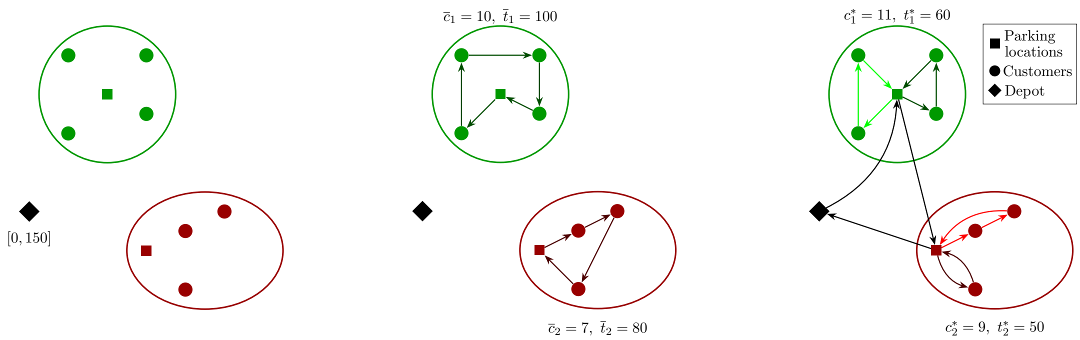
A trade-off between vehicle and deliveryman costs is inherent to the VRPTWMD2R. Figure 2 illustrates it. Figure 2a presents an instance of the problem in which the depot time window closes at instant 150. The best solution considering only the second-level routes cost minimization would be serving each cluster with a single deliveryman, as



(a) Instance. (b) Solution.

Figure 1: An illustrative example of the VRPTWMD2R.

portrayed in Figure 2b. This solution incurs in costs $\bar{c}_1 = 10$ and $\bar{c}_2 = 7$, while the time spent in each cluster is $\bar{t}_1 = 100$ and $\bar{t}_2 = 80$. If these routes were to be taken, these clusters would need to be served by two vehicles since they would not respect the depot time windows when served by a single vehicle. However, if the problem is solved by minimizing all costs, the solution would be the one represented in Figure 2c, in which two deliverymen travel with a single vehicle. The routes inside the clusters are slightly more costly when considered individually and include an additional deliveryman, but they help minimize the overall costs.



(a) Instance. (b) Min deliveryman routes costs. (c) Min overall costs.

Figure 2: A trade-off between deliveryman routes cost and time.

4 Mathematical formulation

We introduce a novel compact mixed-integer programming (MIP) formulation for the VRPTWMD2R. Consider the following parameters:

M_L Maximum number of deliverymen in each vehicle;

f_v Fixed cost associated with each vehicle;

f_d Fixed cost associated with each deliveryman;

c_v Unitary distance cost of first-level routes (vehicles);

c_d Unitary distance cost of second-level routes (deliverymen);

Q Vehicle load capacity;

q_i Demand of cluster $i \in N$;

d_{ij} Distance between first-level nodes i and $j, (i,j) \in A$ (asymmetrical);

t_{ij} Travel time between first-level nodes i and $j, (i,j) \in A$ (asymmetrical);

d^i_{hk} Distance between second-level nodes h and k of cluster $i \in N$, $(h,k) \in A^i$, (asymmetrical);

t^i_{hk} Travel time between second-level nodes h and k of cluster $i \in N, (h,k) \in A^i$ (asymmetrical);

s_h Service time of customer $h \in N^i$ of cluster $i \in N$;

$[a_h, b_h]$ Time window of node $h \in N^i_0$ of cluster $i \in N$.

We define the decision variables taking into account the first- and second-level routes, related to vehicles and deliverymen. Additionally, we need auxiliary variables to model vehicle load and time propagation in the routes. These variables are defined as follows:

x_{ijl} Binary variable that indicates whether a vehicle travels from node i to node j with l deliverymen in a first-level route, $(i,j) \in A, l \in L$;

u_i Vehicle load after leaving node $i \in N_0$;

x^i_{hk} Binary variable that indicates whether a deliveryman travels from node h to node k in a second-level route inside cluster i, $(h,k) \in A^i, i \in N$;

w_h Time when service at node $h \in N^i_0, i \in N$, begins. The arrival time of the vehicle at the parking location of cluster i is represented by w_{0_i} and its departure happens at w_{n_i+1}.

Using the sets, parameters, and decision variables defined so far, we propose the following compact formulation (CF) for the VRPTWMD2R:

$$\text{(CF)}\ \min \sum_{j\in N}\sum_{l\in L}(f_v + lf_d)x_{0jl} + c_v \sum_{(i,j)\in A}\sum_{l\in L} d_{ij}x_{ijl} + c_d \sum_{i\in N}\sum_{(h,k)\in A^i} d^i_{hk}x^i_{hk} \tag{1}$$

$$\text{s.t.} \sum_{i:(i,j)\in A}\sum_{l\in L} x_{ijl} = 1,\ \forall\ j \in N \tag{2}$$

$$\sum_{i:(i,j)\in A} x_{ijl} = \sum_{i:(j,i)\in A} x_{jil},\ \forall\ j \in N, l \in L \tag{3}$$

$$\sum_{i\in N} x_{0il} = \sum_{i\in N} x_{i(n+1)l},\ \forall\ l \in L \tag{4}$$

$$u_j \geq u_i + q_j - Q\left(1 - \sum_{l \in L} x_{ijl}\right), \ \forall\ (i,j) \in A \tag{5}$$

$$\sum_{h:(h,k)\in A^i} x^i_{hk} = 1, \ \forall\ k \in N^i, i \in N \tag{6}$$

$$\sum_{h:(h,k)\in A^i} x^i_{hk} = \sum_{h:(k,h)\in A^i} x^i_{kh}, \ \forall\ k \in N^i, i \in N \tag{7}$$

$$\sum_{h\in N^i} x^i_{0h} = \sum_{h\in N^i} x^i_{h(n_i+1)}, \ \forall\ i \in N \tag{8}$$

$$w_k \geq w_h + s_h + t^i_{hk} - M^i_{hk}(1 - x^i_{hk}), \ \forall\ (h,k) \in A^i, i \in N \tag{9}$$

$$w_{0_j} \geq w_{n_i+1} + t_{ij} - M_{ij}\left(1 - \sum_{l\in L} x_{ijl}\right), \ \forall\ (i,j) \in A \tag{10}$$

$$\sum_{h\in N^j} x^j_{0_j h} \leq \sum_{i:(i,j)\in A} \sum_{l\in L} l x_{ijl}, \ \forall\ j \in N \tag{11}$$

$$u_0 = 0, w_0 = 0 \tag{12}$$

$$x_{ijl} \in \{0,1\}, \ \forall\ (i,j) \in A, l \in L \tag{13}$$

$$q_i \leq u_i \leq Q, \ \forall\ i \in N_0 \tag{14}$$

$$x^i_{hk} \in \{0,1\}, \ \forall\ (h,k) \in A^i, i \in N \tag{15}$$

$$a_h \leq w_h \leq b_h, \ \forall\ h \in N^i_0, i \in N_0. \tag{16}$$

The objective function (1) seeks to minimize the total fixed costs of both vehicles and deliverymen and the distance costs of both vehicle and deliveryman routes. Constraints (2) ensure that every cluster is visited by exactly one vehicle. Constraints (3) and (4) are flow conservation constraints for first-level routes. Constraints (5) control the load flow in vehicle routes. Constraints (6)–(8) are similar to (2)–(4) but considering second-level routes. Constraints (9) and (10) control the time propagation for deliveryman and vehicle routes, respectively. Beyond defining the arrival time at each customer, constraints (9) implicitly define the time spent in each cluster since they involve the moments that the deliverymen depart from and arrive at the parking locations. Constraints (10) use this information to synchronize the first- and second-level routes by defining that the deliverymen start to serve a cluster j after having served a cluster i and having traveled to cluster j if they travel in a vehicle that goes from i to j. In these constraints, we define $M^i_{hk} = \max\{0, b_h + s_h + t^i_{hk} - a_k\}$ and $M_{ij} = \max\{0, b_{n_i+1} + t_{ij} - a_{0_j}\}$ as the smallest possible values to ensure that the constraints are valid. Constraints (11) also couple the first- and second-level routes by defining that the number of deliveryman routes inside a cluster is, at most, the number of deliverymen that arrive at it (it is possible that not all deliverymen visiting a cluster leave the vehicle). Constraints (12)–(16) define the domain of the decision variables.

4.1 Valid inequalities

Formulation CF can be strengthened by the following valid inequalities (VIs) to improve its linear relaxation. In these constraints, let $e_{il}, i \in N, l \in L$, be a lower bound on the time needed to serve cluster i with l deliverymen and $m_i, i \in N$, be a lower bound on the number of deliverymen needed to serve cluster i feasibly.

$$\sum_{h \in N^i} x^i_{0_i h} \geq 1, \; \forall \; i \in N \tag{17}$$

$$\sum_{(h,k) \in A^i : h,k \in S} x^i_{hk} \leq |S| - 1, \; \forall \; S \subset N^i, i \in N : |S| \in \{2, 3\} \tag{18}$$

$$x^i_{hk} = 0, \; \forall \; (h,k) \in A^i, i \in N : (a_h + s_h + t^i_{hk} > b_k) \tag{19}$$

$$\sum_{j \in N} \sum_{l \in L} x_{0jl} \geq \left\lceil \frac{1}{Q} \sum_{i \in N} q_i \right\rceil \tag{20}$$

$$\sum_{(i,j) \in A : i,j \in S} \sum_{l \in L} x_{ijl} \leq |S| - 1, \; \forall \; S \subset N : |S| \in \{2, 3\} \tag{21}$$

$$x_{ijl} = 0, \; \forall \; (i,j) \in A : (q_i + q_j > Q) \vee (a_{n_i+1} + t_{ij} > b_{0_j}) \tag{22}$$

$$w_{n_i+1} \geq w_{0_i} + \sum_{j : (i,j) \in A} \sum_{l \in L} e_{il} x_{ijl}, \; \forall \; i \in N \tag{23}$$

$$x_{ijl} = 0, \; \forall \; (i,j) \in A, l \in L : (m_i > l) \vee (m_j > l) \tag{24}$$

$$\sum_{h \in N^i} x^i_{0_i h} \geq m_i, \; \forall \; i \in N \tag{25}$$

Constraints (17)–(22) are common in the literature, while constraints (23)–(25) are novel VIs proposed specifically for this problem. Constraints (17) ensure that at least one deliveryman leaves each parking location. Constraints (18) eliminate small subtours of two and three customers in second-level routes. Constraints (19) remove infeasible second-level arcs due to time window incompatibility. Constraints (20) define a lower bound on the number of vehicles needed to serve all the clusters based on the total cluster demands and vehicle capacity. Constraints (21) eliminate subtours for sets of two and three clusters in first-level routes. Constraints (22) eliminate first-level arcs that are infeasible due to vehicle capacity or time windows incompatibility. Constraints (23) provide an estimation on the minimum time spent on the cluster. Constraints (24) forbid the visit of the cluster by a vehicle with fewer deliverymen than needed to serve it. Constraints (25) ensure that the number of deliverymen leaving a parking location respects its lower bound. Since $m_i \geq 1, \; \forall \; i \in N$, constraints (17) are redundant when constraints (25) are considered. Hence, either constraints (17) or (25) are included, never both.

On top of these constraints, time windows are tightened based on the earliest arrival time from the depot and the latest departure time to arrive while the depot is still open (Ascheuer, Fischetti, and Grötschel, 2001).

5 Benders decomposition

Since the definition of the deliveryman routes depends on the vehicle routes and the number of deliverymen serving each cluster, the CF can be decomposed in a Benders fashion (Benders, 1962; Hooker and Ottosson, 2003; Codato and Fischetti, 2006). This way, the master problem (MP) defines the first-level routes and the number of deliverymen in each vehicle, and the subproblem (SP) defines the second-level routes.

To exploit this characteristic of the VRPTWMD2R and efficiently solve it, we develop an exact algorithm based on a branch-and-Benders-cut (BBC) scheme (Moreno, Munari, and Alem, 2019; Moreno, Munari, and Alem, 2020). To this extent, we improve the Benders decomposition by including valid inequalities and developing lower bounding techniques. Section 5.1 presents the MP, Section 5.2 defines the SP, Section 5.3 introduces useful lower bounds, and Section 5.4 discusses the BBC algorithm.

5.1 Master Problem

Let $\eta_i, i \in N$, be a variable representing the cost of the deliveryman routes inside cluster i with a lower bound $\underline{\eta}_i \geq 0$. Let R be the set of all pairs (r, l) of vehicle routes r and number of deliverymen l that are feasible given first- and second-level constraints; and $\overline{R}$ be the set of pairs (r, l) that are feasible considering first-level constraints (information in the MP), but infeasible considering second-level constraints (information in the SP). It is clear that $R \cap \overline{R} = \emptyset$.

Let N_r be the set of clusters visited by route r and A_r be the set of arcs of route r. Given a pair $(r, l) \in R$, let $g_{rli}, i \in N$, represent the cost of deliveryman routes inside cluster i when visited by a vehicle traveling with l deliverymen along route r, and $c_{rl} = \sum_{i \in N_r} g_{rli}$ be the sum of these costs throughout the vehicle route.

Given these definitions, the CF can be reformulated as the following MP:

$$\text{(MP)} \min \sum_{j \in N} \sum_{l \in L} (f_v + l f_d) x_{0jl} + c_v \sum_{(i,j) \in A} \sum_{l \in L} d_{ij} x_{ijl} + \sum_{i \in N} \eta_i \tag{26}$$

$$\text{s.t. } (2)\text{–}(5),\ (10),\ (12)\text{–}(14)$$

$$\sum_{i \in N_r} \eta_i \geq c_{rl} \left(\sum_{(i,j) \in A_r} x_{ijl} - |A_r| + 1 \right),\ \forall\ (r, l) \in R \tag{27}$$

$$\sum_{(i,j) \in A_r} x_{ijl} \leq |A_r| - 1,\ \forall\ (r, l) \in \overline{R} \tag{28}$$

$$a_i \leq w_i \leq b_i,\ \forall\ i \in \{0_h, n_h + 1\}, h \in N \tag{29}$$

$$\eta_i \geq \underline{\eta}_i,\ \forall\ i \in N. \tag{30}$$

The objective function (26) is equivalent to (1) with a different form of calculating the deliveryman routes cost. Constraints (27) correspond to the so-called optimality cuts, which define the cost of second-level routes inside the clusters visited by a vehicle traveling along a first-level route r and carrying l deliverymen. Constraints (28) consist in the so-called feasibility cuts, removing from the set of feasible solutions of the MP the vehicle routes that are infeasible due to the corresponding deliveryman routes. Constraints (29) define the time windows of parking locations, and constraints (30) establish a lower bound on the cost of deliveryman routes inside each cluster. The MP can be further strengthened by VIs (20)–(24). We shall refer to the MP without the feasibility and optimality cuts as the relaxed MP (RMP).

Constraints (27) and (28) are based on the traditional route-based optimality and feasibility cuts. However, we propose using the path cuts introduced by Parada et al. (2023), in which the first-level route arcs that are connected to the depot are removed from the cut. Propositions 1 and 2 ensure the validity of this approach for the VRPTWMD2R. Proposition 3 includes an additional summation in $l \in L$ in the feasibility cuts. These modifications yield better cuts that help boost the algorithm's performance. To this extent, we denote by $\hat{A}_r \subset A_r$ the set of arcs in route r without those connected to the depot.

Proposition 1. *The constraints*

$$\sum_{i \in N_r} \eta_i \geq c_{rl} \left(\sum_{(i,j) \in \hat{A}_r} x_{ijl} - |\hat{A}_r| + 1 \right), \ \forall \ (r,l) \in R \tag{31}$$

can replace constraints (27) *as valid optimality cuts if* $|N_r| > 1$ *and the triangular inequality holds for vehicle routes.*

Proof. Given a pair $(r,l) \in R$ with $|N_r| > 1$, let $r = (0, r_1, r_2, \ldots, r_{|N_r|}, n+1)$ be the sequence of nodes visited in first-level route r. Let us define path $p = (r_1, r_2, \ldots, r_{|N_r|})$ as the path of $|N_r|$ clusters visited in route r. With these definitions, $\hat{A}_r$ can be interpreted as the set of arcs of p. Hence, constraints (31) state that, for every first-level route that contains path p, the cost of second-level routes inside the clusters of path p is at least c_{rl}, i.e., the cost of traveling the path in a vehicle route that does not visit any cluster out of the path. This is true because in every first-level route $\overline{r} \supset p, \overline{r} \neq r$, there are clusters visited before and/or after path p, making the dynamic of the deliverymen inside the clusters of path p more constrained than in route r, as triangular inequality holds. Since it is more constrained, the costs of the deliveryman routes in the clusters of path p is at least c_{rl}, proving the validity of constraints (31) as optimality cuts. □

Proposition 2. *The constraints*

$$\sum_{(i,j) \in \hat{A}_r} x_{ijl} \leq |\hat{A}_r| - 1, \ \forall \ (r,l) \in \overline{R} \tag{32}$$

can replace constraints (28) *as valid feasibility cuts if the triangular inequality holds for vehicle routes.*

Proof. Following the notation used on the proof of Proposition 1, constraints (32) state that $(r, l) \in \overline{R} \Rightarrow (\overline{r}, l) \in \overline{R},\ \forall\ \overline{r} \supset p$, i.e., if a first-level route $r = (0, p, n+1)$ is infeasible when traveled by a vehicle with l deliverymen, every other route $\overline{r} \supset p$ will also be infeasible when traveled with the same number l of deliverymen. This is true because, if the triangular inequality holds, including any cluster before or after path p would make the second-level routes inside the clusters of p more constrained than in route r. If these deliveryman routes are infeasible without this additional cluster, they will remain as such with this addition. □

Proposition 3. *The constraints*

$$\sum_{(i,j)\in \hat{A}_r} \sum_{\bar{l}\in L:\bar{l}\leq l} x_{ij\bar{l}} \leq |\hat{A}_r| - 1,\ \forall\ (r, l) \in \overline{R} \tag{33}$$

can replace constraints (28) *as valid feasibility cuts if the triangular inequality holds for vehicle routes.*

Proof. It is true that $(r, l) \in \overline{R} \Rightarrow (r, \bar{l}) \in \overline{R},\ \forall\ \bar{l} \in L, \bar{l} < l$, because reducing the number of deliverymen on a first-level route makes the second-level routes inside the clusters more constrained. Thus, if the first-level route is infeasible with l deliverymen, it will also be with $\bar{l} < l$. Therefore, given Proposition 2,

$$\sum_{(i,j)\in \hat{A}_r} x_{ij\bar{l}} \leq |\hat{A}_r| - 1,\ \forall\ \bar{l} \leq l, (r, l) \in \overline{R}$$

are valid feasibility cuts if the triangular inequality holds. By constraints (2)–(4), at most one value of l is associated with a vehicle route r, allowing for the summation in $\bar{l}$ that yields constraints (33) as valid feasibility cuts. □

Note that it is possible to aggregate the optimality cuts (31) by summing them up for all number of deliverymen $\bar{l} < l$, as we did for feasibility cuts (33). However, preliminary results indicate that, in the case of optimality cuts, this is only beneficial for small instances, and has a negative effect for medium and large instances as the cuts become too dense. Therefore, we use the disaggregated version as presented above.

Comparing the improved path cuts (31) and (33) with the original route cuts (27) and (28), it is clear that the improved versions yield stronger LP relaxations. Furthermore, while each route cut is active in a single integer solution, the improved versions are active in more than one solution. This justifies the improvements from a theoretical perspective. Our experiments confirm that this theoretical improvement is translated into a better performance of the BBC, as shown in Section 6.3.

5.2 Subproblem

To generate optimality and feasibility cuts we resort to an SP that optimizes the cost of the deliveryman routes for each pair $(r, l) \in R$, or determines that it is infeasible to perform first-level route r with l deliverymen if $(r, l) \in \overline{R}$. Given a pair (r, l), the corresponding SP is defined by

$$\text{(SP)} \min c_d \sum_{i \in N_r} \sum_{(h,k) \in A^i} d^i_{hk} x^i_{hk} \tag{34}$$

$$\text{s.t.} \quad \sum_{h:(h,k) \in A^i} x^i_{hk} = 1, \ \forall \ k \in N^i, i \in N_r \tag{35}$$

$$\sum_{h:(h,k) \in A^i} x^i_{hk} = \sum_{h:(k,h) \in A^i} x^i_{kh}, \ \forall \ k \in N^i, i \in N_r \tag{36}$$

$$\sum_{h \in N^i} x^i_{0_i h} = \sum_{h \in N^i} x^i_{h(n_i+1)}, \ \forall \ i \in N_r \tag{37}$$

$$w_k \geq w_h + s_h + t^i_{hk} - M^i_{hk}(1 - x^i_{hk}), \ \forall \ (h, k) \in A^i, i \in N_r \tag{38}$$

$$\sum_{h \in N^i} x^i_{0_i h} \leq l, \ \forall \ i \in N_r \tag{39}$$

$$w_{0_j} \geq w_{n_i+1} + t_{ij}, \ \forall \ (i, j) \in A_r \tag{40}$$

$$x^i_{hk} \in \{0, 1\}, \ \forall \ (h, k) \in A^i, i \in N_r \tag{41}$$

$$a_h \leq w_h \leq b_h, \ \forall \ h \in N^i_0, i \in N_r \cup \{n + 1\}. \tag{42}$$

The objective function (34) seeks to minimize the total cost of second-level routes. Constraints (35)–(38) are equivalent to (6)–(9) but restricted to the nodes in N_r. Constraints (39) limit the number of deliveryman routes inside a cluster to the number of deliverymen traveling in the vehicle route r. Constraints (40) define the vehicle time flow, i.e., the deliverymen leave a parking location (w_{0_j}) after serving the previous cluster in the route (w_{n_i+1}) and traveling from one cluster to the next one in the vehicle route (t_{ij}). Finally, constraints (41) and (42) define the domain of the decision variables.

It is important to notice that this SP comes from splitting a solution in routes and is, therefore, separable by vehicle route r, but not by deliveryman routes in each cluster due to the trade-off between deliveryman routes cost and time discussed in Section 3. There is a time dependency among different clusters served by the same vehicle given by constraints (40). Thus, although there might be a short deliveryman route to serve a given cluster's customers, if this route takes a long time it might affect the feasibility of the corresponding vehicle route by not respecting the next cluster's time window. Hence, in this case, it would be necessary to take longer deliveryman routes that would be more costly but feasible considering the vehicle route to be followed.

The SP can be strengthened by VIs (17)–(19), and (25). We also define the following VIs for the SP relative to a pair $(r, l) \in R \cup \overline{R}$:

$$w_{n_i+1} \geq w_{0_i} + e_{il}, \ \forall \ i \in N_r, \tag{43}$$

which defines a lower bound on the time spent in each cluster of the vehicle route considering the number of deliverymen traveling in it.

Finally, time windows are tightened. Ascheuer, Fischetti, and Grötschel (2001) propose to tighten the time windows based on all of the possible predecessors and successors of a node. Since in the SP the vehicle route is predefined, each cluster has a unique predecessor and a unique successor, making this tightening very efficient.

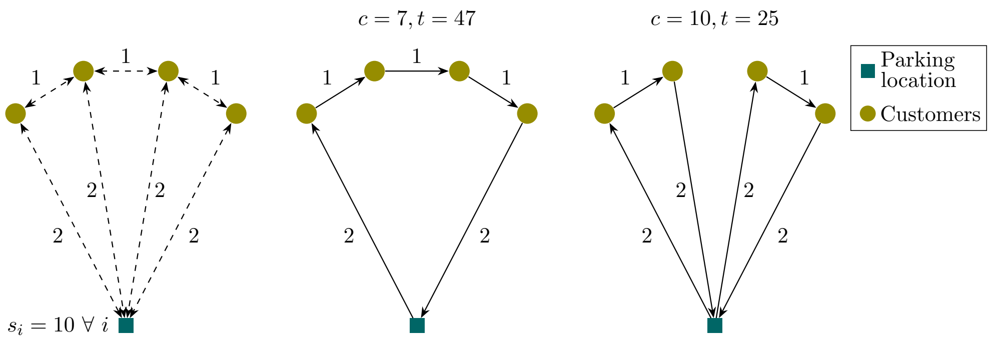
5.3 Lower bounds

The definition of the MP relies on the lower bound $\underline{\eta}_i, i \in N$, for the cost of the deliveryman routes inside a cluster i. Moreover, VIs (23) and (43) depend on the lower bound $e_{il}, i \in N, l \in L$, for the time spent in cluster i when served with l deliverymen; and VIs (24) and (25) are based on a lower bound $m_i, i \in N$, for the number of deliverymen needed to serve a cluster. To tightly define these lower bounds, we solve a sequence of MIP models based on the SP defined for a vehicle route that goes from the depot to a cluster $i \in N$ and then back to the depot. For calculating $\underline{\eta}_i$, the SP is solved for every cluster $i \in N$ defining $l = M_L$ in constraints (39). For defining e_{il}, the SP is solved by changing its objective function to $w_{n_i+1} - w_{0_i}$, for every node $i \in N$ and deliverymen number $l \in L$. The value of m_i is assessed by the feasibility of the MIP model solved for time minimization. If this model is infeasible for a given l, then $m_i \geq l + 1$. Otherwise, if it is feasible for every $l \in L$, then $m_i = 1$. When solving these MIP models, we define a lower bound on the time spent in cluster $i \in N$ when served by $l \in L$ deliverymen as

$$\max \left\{ \frac{1}{l} \sum_{h \in N^i} s_h, \max_{h \in N^i} \left\{ t^i_{0_i h} + s_h + t^i_{h(n_i+1)} \right\} \right\}.$$

Notably, even though distances and travel times are proportional, the cost and time minimization MIP models yield different solutions due to the customers time windows and the possibility of serving the clusters with more than one deliveryman. This difference has already been explained and is illustrated in Figure 3 by showing a cluster with four customers. Figure 3a presents the cluster data, indicating the cost of each arc and that each customer has a service time of 10 units (arcs cost, distance, and travel time are equivalent in the picture). If there are two available deliverymen in this cluster, the cost minimization MIP model would use only one of them to produce the second-level route portrayed in Figure 3b, since it is the shortest option with a total cost of 7 and total time of 47. Nevertheless, if the goal is to minimize the time spent in the cluster, using both deliverymen traveling the routes shown in Figure 3c would be the best choice, since the customers would be served in parallel, yielding a solution with total cost 10 and total time 25. Hence, it is necessary to solve different MIP models for each lower bound. This is partly what creates the trade-off between deliveryman routes cost and time discussed in Section 3.

Calculating these lower bounds requires $n(M_L + 1)$ runs of the MIP models of the SPs. Although computationally burdensome, this evaluation significantly improves the performance of the algorithms, as shown in Section 6.



(a) Instance. (b) Cost minimization. (c) Time minimization.

Figure 3: Different solutions by minimizing deliveryman routes cost or time.

5.4 Branch-and-Benders-cut

Given the exponential number of optimality and feasibility cuts, it is impractical to enumerate all of them a priori. Instead, the best approach is to solve the RMP and include optimality and feasibility cuts as needed in a BBC fashion (Moreno, Munari, and Alem, 2019). To this extent, we solve the MP using a branch-and-cut algorithm that starts with the RMP and, every time a feasible integer solution to the RMP is found, we evaluate the corresponding SPs. If the solution of the RMP respects the optimality and feasibility cuts, we update the incumbent solution (if the new solution is better than the incumbent), otherwise we include the corresponding optimality and feasibility cuts.

The following steps represent the BBC algorithm:

1. Define cost and time lower bounds on the deliveryman routes in each cluster (Section 5.3);

2. Define the initial RMP and start the branch-and-cut method (Section 5.1);

3. Every time a feasible solution of the RMP is found in the branch-and-cut tree, separate the solution by vehicle routes, tighten the clusters time windows considering the vehicle route serving them, and solve the SPs (Section 5.2). For each SP, if it is feasible, include the corresponding optimality cuts (31), otherwise include the corresponding feasibility cuts (33). If all of the SPs are feasible, and the solution cost updated with the deliveryman routes cost is lower than the incumbent cost, update the incumbent.

The algorithm terminates once all of the nodes of the branch-and-cut tree have been processed.

6 Computational experiments

We now describe the computational experiments performed to assess the performance of the proposed model and algorithms and their suitability to solve the VRPTWMD2R.

The approaches were implemented in C++ and use Gurobi 10.0.2 with an optimality gap tolerance of 10^{-7}. The experiments were run on computers equipped with 2xAMD Rome 7532 processors running at 2.4GHz and up to 64GB of RAM for the CF, and 32GB for the BBC, with a time limit of 7,200s. For the MIP models that determine the lower bounds described in Section 5.3 we set a time limit of 10s; when the solver was unable to prove optimality within this time limit, we used the lower bound obtained by the solver to define the lower bounding parameter on time or cost. All instances and detailed results are available at `https://www.dep.ufscar.br/munari/vrptwmd/`.

Section 6.1 describes the instances used in our experiments. In Section 6.2, we present the results obtained with the CF and the different sets of VIs, allowing us to assess the effectiveness of the existing and new VIs. In Section 6.3, we discuss the results obtained with the BBC method. Finally, Section 6.4 provides managerial insights for this practical problem.

6.1 Instances

The generated instances are based on the Solomon (1987) instances for the VRPTW from classes C1, R1, and RC1. We considered that each node in a Solomon instance represents the parking location of a cluster in the VRPTWMD2R. Then, we generated one to seven customer locations around each parking location to create the customers in the corresponding cluster. Coordinates of the customers were generated following a normal distribution with mean in the parking location's coordinates and standard deviation $\sigma = 3$, which showed to be well suited for the problem representation. In the Solomon instances, only some nodes have time windows; if they do, i.e., the parking location has a time window, then time windows were generated for the customers assigned to them. These time windows were randomly generated considering the time window opening of the cluster and the average width of the clusters time windows, while ensuring feasibility of the instances. The service time of each customer is assumed to be the same as that of the corresponding parking locations.

We generated instances of five different sizes, namely 10–40, 15–60, 15–85, 20–80, and 25–125, in which the first number represents the number of clusters (parking locations) and the second number represents the total number of customers. This way, there are instances with 50, 75, 100, and 150 nodes, which are realistic for many last-mile logistics applications. There are 29 instances of each size, for a total of 145, all available online.

Following Pureza, Morabito, and Reimann (2012), we defined the cost parameters as $(f_v, c_v, f_d, c_d) = (1000, 10, 100, 1)$ and allowed up to $M_L = 3$ deliverymen per vehicle. The distances were calculated assuming Euclidean distances truncated to integers. For the vehicles, distance and travel time were considered equivalent, and deliverymen were assumed to travel at one-third of the vehicles' speed. After calculating distances and travel times, the Floyd-Warshall algorithm (Cormen et al., 2009) was run to ensure the triangular inequality was valid.

6.2 Compact formulation and valid inequalities

We first assess the performance of our CF (1)–(16), of the existing VIs (17)–(22), and of the newly proposed VIs (23)–(25). Table 1 shows the summarized results of the experiments with the CF and VIs (detailed results are provided as supplementary material). It presents the results for the CF only (hereinafter referred to as CF1), the CF enhanced with VIs (17)–(22) from the literature (CF2), and the CF enhanced with VIs (18)–(25), both novel and literature-based (CF3). As discussed in Section 4, VIs (17) are redundant when VIs (25) are considered and are, therefore, not included in the latter scenario. In this table, "LR" stands for "LP relaxation", "LB" for "lower bound", "UB" for "upper bound", "Gap" for the optimality gap provided by the solver (as a percentage), "Time" for the running time in seconds, and "Opt" for the number of instances for which the solver has proved optimality for the corresponding model. The values of LR, LB, UB, Gap, and Time represent the corresponding average. We present the average gap as the average of optimality gaps of instances, not the gap calculated with the average LB and UB.

	Size	LR	LB	UB	Gap (%)	Time (s)	Opt
CF1	10–40	1,027	5,508	7,138	26.93	6,069	5
	15–60	1,354	6,912	10,453	34.42	6,704	2
	15–85	1,398	6,900	12,403	46.47	6,954	1
	20–80	1,764	9,088	14,614	38.96	6,723	2
	25–125	2,071	11,446	20,426	45.63	6,954	1
	Total	1,523	7,971	13,007	38.48	6,681	11
CF2	10–40	4,063	5,727	7,170	21.51	6,081	7
	15–60	5,752	7,274	10,447	30.14	6,470	3
	15–85	6,023	7,379	12,255	40.50	6,954	1
	20–80	7,515	9,411	14,462	35.87	6,723	2
	25–125	10,066	12,017	20,220	41.62	6,953	1
	Total	6,684	8,362	12,911	33.93	6,636	14
CF3	10–40	4,637	7,131	7,131	0.00	316	28
	15–60	6,445	10,189	10,228	0.68	2,262	26
	15–85	7,757	12,018	12,116	1.26	4,108	13
	20–80	8,589	13,499	13,988	4.55	4,184	14
	25–125	12,653	18,358	19,195	5.76	5,155	9
	Total	8,016	12,239	12,532	2.45	3,205	90

Table 1: Results of the experiments with CF and different sets of VIs.

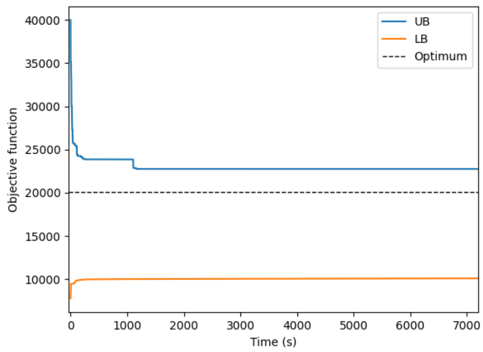
These results indicate that the VIs significantly strengthen the LP relaxation of the CF. The inclusion of the VIs from the literature improves the average value of the LR in 338.92% and the novel VIs provide an additional improvement of 19.94%, leading to a total increase of 426.42% in the LR values. Moreover, for instances with sizes 15–85 and 25–125, the value of the LR of CF3 is higher than the final LB obtained after running the solver for two hours with the other two model configurations.

Regarding the performance of the MIP solver, the CF1 yields poor results, with high gaps even for the smallest instances. The solver proved optimality on few instances (11 out of a total of 145). The VIs from the literature (CF2) improve its performance, especially by lifting the average LB in 4.91%, which yields a modest 4.55% improvement in the average gap. They also help prove optimality for three other instances, reaching 9.66% of the instances (14/145). Still, the average gap is 21.51% for the smallest-sized instances.

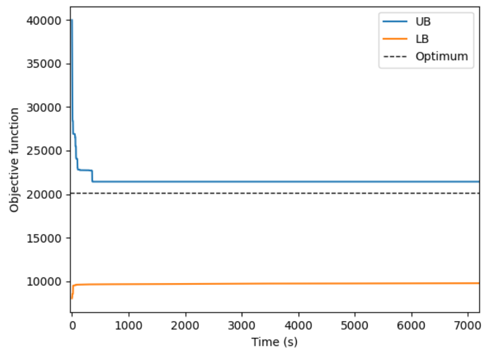
The combination of the VIs from the literature with the VIs proposed for the problem (CF3) produces a significant improvement in the results, leading to an additional 46.36% increase in the average LB and 31.48% reduction in the average gap. Furthermore, the number of instances with proven optimality increases to 90, which is more than half of the total instances, and more than six times the number of instances proved to optimality before. The runtime is significantly improved as a consequence of the new VIs and their effect in proving optimality. Note that the small instances can now be solved in about five minutes, and the average runtime is decreased by more than half.

Figure 4 shows the convergence curves of the different CFs when solving instance R110 with size 25–125, which illustrates a common behavior of these models in many instances. Figures 4a and 4b indicate that both the CF1 and the CF2 start from high UBs that rapidly decrease and the LB increases a little in the first few seconds. However, after 1000s of runtime, there is little improvement either in the UB or the LB, leading to large gaps (55.67% for the CF1 and 54.42% for the CF2). The CF3, as portrayed in Figure 4c, starts a few seconds later because it calculates the lower bounds discussed in Section 5.3 before starting the solution procedure. As in the other approaches, the UB rapidly decreases, but the difference here is the significant increase in the LB right in the first seconds of runtime. This figure illustrates the effect shown in Table 1. Indeed, although the improvement in the LR from using the novel VIs is small compared to the VIs from the literature, it significantly helps the performance of the MIP solver by increasing the LB throughout the branch-and-cut search tree. Nevertheless, these improvements do not overcome the tailing-off effect shown by the CF1 and the CF2, preventing the algorithm from proving optimality within the time limit, and finishing with an optimality gap of 5.80%. It is worth mentioning that, to assess whether longer runtimes would allow the solver to prove optimality for this instance, we have run the CF3 solving this specific instance with a time limit of twenty hours and, even though the gap was reduced, it was not possible to prove optimality.

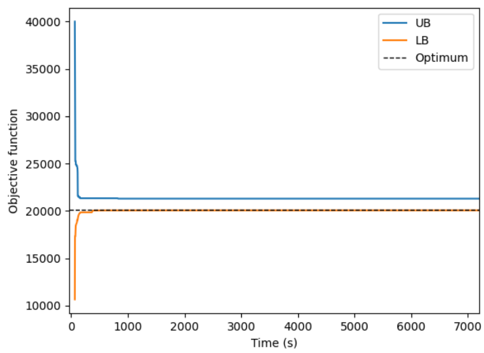
These analyses have demonstrated the added value of the VIs from the literature and the significant improvement obtained with the newly proposed VIs for our problem. Using the CF3, the solver proved optimality for many instances and provided good bounds for the remaining larger instances. This version of the model is used in the next section to assess the performance of our BBC algorithm.



(a) CF1.



(b) CF2.



(c) CF3.

Figure 4: Convergence curves for instance R110 with size 25–125.

6.3 Branch-and-Benders-cut algorithm

Since the previous experiments clearly show the efficiency of the proposed VIs, in our BBC method the RMP always includes the VIs (20)–(24), and the SP includes VIs (17)–(19), (25), and (43).

The first experiments with the BBC method evaluate the relevance of the cut improvements discussed in Section 5.1. Table 2 presents the results in a subset of instances (even numbered instances) considering two different versions of the method: with route cuts (27) and (28); and with improved path cuts (31) and (33). Notably, the performance with the improved cuts is slightly worse for smaller instances but it is significantly better for larger ones. The number of instances to which the versions have proved optimality is the same and hence was not shown. On average, the improved cuts yield positive impacts in the LB and UB, leading to a 0.26% gap improvement and 10% time reduction. Given these results, the remaining experiments with the BBC are all run with the path cuts (31) and (33).

	Size	LB	UB	GAP (%)	Time (s)
BBC with route cuts (27) and (28)	10 – 40	6,922	6,922	0.00	20
	15 – 60	10,039	10,098	1.02	1,097
	15 – 85	11,874	11,957	1.10	656
	20 – 80	12,773	13,764	7.97	3,143
	25 – 125	17,394	19,016	8.93	3,115
	Total	11,800	12,351	3.80	1,606
BBC with improved cuts (31) and (33)	10 – 40	6,922	6,922	0.00	22
	15 – 60	10,031	10,098	1.16	758
	15 – 85	11,873	11,957	1.12	632
	20 – 80	12,844	13,764	7.58	3,163
	25 – 125	17,382	18,717	7.87	2,672
	Total	11,810	12,291	3.54	1,449

Table 2: Impact of cut improvements in the BBC method.

Table 3 presents the results of the experiments with the BBC algorithm compared with the CF3. Figure 5 shows the convergence curves for instance R110 with size 25–125. In addition to the LB and UB curves, this figure also shows the points in which optimality cuts were inserted.

Compared to the CF3, the BBC algorithm reduces another 0.06% in the average gap and gives a slight improvement in the average UB for large instances of sizes 20–80 and 25–125. The greatest improvements, however, are in the number of instances solved to proven optimality and in the average runtime.

The BBC proved optimality for all instances with sizes 10–40, and for 28 out of 29 instances of sizes 15–60 and 15–85. In total, it proved optimality for 129 instances, which represents 88.97% of the total number of instances, and an increase of 43.33% compared to the CF3. Even for the instances with sizes 20–80 and 25–125, to which there was no

	Size	LB	UB	Gap (%)	Time (s)	Opt
CF3	10–40	7,131	7,131	0.00	316	28
	15–60	10,189	10,228	0.68	2,262	26
	15–85	12,018	12,116	1.26	4,108	13
	20–80	13,499	13,988	4.55	4,184	14
	25–125	18,358	19,195	5.76	5,155	9
	Total	12,239	12,532	2.45	3,205	90
BBC	10–40	7,131	7,131	0.00	18	29
	15–60	10,196	10,228	0.56	435	28
	15–85	12,075	12,116	0.54	374	28
	20–80	13,425	13,972	4.89	2,003	22
	25–125	18,205	19,083	5.96	2,110	22
	Total	12,207	12,506	2.39	988	129

Table 3: Results of the experiments with the best versions of the CF and BBC approaches.

improvement in the LB and gap when comparing the BBC with the CF3, the number of instances solved to proven optimality went from 14 and 9 to 22 and 22 with the BBC. For these sizes, the average LB and gap did not improve because the BBC performed worse than the CF3 in a few instances, despite being superior in most of them.

Moreover, the runtime was drastically reduced. Small instances were solved to optimality within seconds by the BBC method, and the average runtime, which was close to 2 hours for the CF1 and close to 1 hour for the CF3, was reduced to slightly more than 15 minutes. In part, this improvement is caused by the overcoming of the tailing-off effect, as shown in Figure 5. The BBC method proved optimality for that instance in less than 200s, while the other approaches have high gaps after 7200s and, as discussed in Section 6.2, could not prove optimality even after twenty hours of runtime. This leads to significant improvements in the runtime of the algorithm, with the average value representing 69.17% of reduction compared to the results of the CF3. In the instances of size 10–40, the runtime reduction is of 94.30%. This result is especially important considering that exact methods usually suffer from being very time-consuming, while the proposed BBC has presented reasonable running times for most instances.

Table 4 provides a closer look at the cuts inserted in the BBC algorithm. The number of feasibility cuts is less than one per instance on average. For the instances with size 15–60, no feasibility cut was needed in any instance. This indicates a very good performance of the proposed lower bounds for ensuring feasibility of the solution provided by the RMP. It also shows that many instances do not need feasibility cuts, as the one portrayed in Figure 5. The number of inserted optimality cuts grows with the instance sizes, but there are fewer than two cuts for each node on average. Figure 5 illustrates the fact that when a new optimality cut is inserted, a new incumbent solution is often found, reducing the UB value.

Regarding the time spent separating these cuts, it grows rapidly with the instance sizes and is directly related to the number of cuts added. Additionally, each cut separation takes

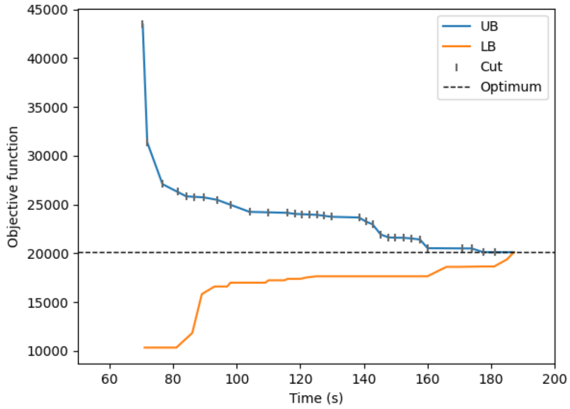


Figure 5: Convergence of the BBC method for instance R110 with size 25–125.

Size	# of feasibility	# of optimality	Total separation (s)
10–40	0.21	38.72	7.71
15–60	0.00	94.93	31.00
15–85	0.69	87.17	95.67
20–80	1.17	154.28	57.45
25–125	1.79	222.03	437.76
Avg	0.77	119.43	125.92

Table 4: Average number of cuts and separation times in the BBC algorithm.

longer for larger instances as they have more customers in each route and larger clusters.

When comparing the results of our BBC approach with those obtained by solving the CF alone (CF1), the BBC yields a 36.09% reduction in the average gap, the number of instances solved to optimality is increased by 1,172.73%, the average UB is improved by 3.85%, the average LB increases 53.14%, and the average runtime is reduced in 85.21%, which highlights the suitability of the proposed method to solve the problem.

Moreover, when looking at the results of the algorithms considering different instance sizes, it can be seen that it becomes more challenging to solve the problem as the instances grow. Nonetheless, different solution methods may be more or less sensitive to this increase in the difficulty in solving the problem depending if the size changes more expressively in the number of customers or clusters. Instances with sizes 15–85 and 20–80, for example, have a total of 100 nodes. On the one hand, CF1 and CF2 have better performances for instances with size 20–80 than for instances with size 15–85, indicating that the size of clusters affects these approaches more than the number of clusters. On the other hand, CF3 and BBC have better performances in the instances with size 15–85 than in those with size 20–80, suggesting that these methods are more affected by the number of clusters than by the cluster size. This shows that the proposed BBC method and the novel VIs were effective in decreasing the difficulty related to the second-level routes, as was our goal with those approaches given that these routes are less relevant (much cheaper and highly dependent on the other decisions) than the first-level routes and the number of vehicles or deliverymen used. Furthermore, the number of cuts added in the BBC for the instances with size 20–80 is 76.99% higher than for the instances with size 15–85, even though the separation time is 39.95% lower.

6.4 Managerial insights

We ran experiments to assess the relevance of considering deliveryman routes and to perform sensitivity analysis on the results. Experiments were run with the BBC method in a subset of instances of sizes 15–85, 20–80, and 25–125 to which this method proved optimality in all configurations.

As discussed in Section 2, the previous works on the VRPTWMD ignored the deliveryman routes by considering that deliverymen have limited capacity and thus cannot visit more than one customer without returning to the vehicle. We adapted our methods to consider this alternative of having the deliverymen perform round trips to all customers in the cluster by simply setting the distance $d_{ij} = d_{i0} + d_{0j}$, where i and j are two customers and 0 represents the parking location. This new distance matrix effectively models the case of round trips to each customer. The travel times were defined accordingly. The results presented in Table 5 contrast this situation with the VRPTWMD2R proposed in this paper.

	Ignoring deliveryman routes	With deliveryman routes
# of vehicles	6.78	6.22
# of deliverymen	18.39	16.11
First-level distance	454.22	423.00
Second-level distance	669.00	474.56
Total cost	13,827.89	12,537.89

Table 5: The importance of considering deliveryman routes.

In spite of being a problem much easier to solve (the solution times were roughly one-third), ignoring the second-level routes creates significantly worse results. Since it overestimates the deliveryman routes time and distance, it has a greater need for both vehicles and deliverymen. The overall costs are 10.29% higher, highlighting the importance of considering the deliveryman routes in the problem.

These results also highlight that savings are expected if deliverymen can perform small routes instead of visiting one customer at a time. In applications where walking deliverymen cannot carry goods to serve more than one customer at a time, small scooters or cargo bikes can enable this. More generally, this analysis sheds light on the limitations and benefits of drone delivery, depending on the drone capacity and range.

Another important assessment is the trade-off between vehicle and deliveryman costs discussed in Section 3. Table 6 compares the results for three different cost structures, in which the first- and second-level cost components in (f_v, c_v, f_d, c_d) are set as follows:

(i) deliverymen ten times cheaper than vehicles $(1000, 10, 100, 1)$; (ii) deliverymen and vehicles with the same costs $(100, 1, 100, 1)$; and (iii) deliverymen ten times more expensive than vehicles $(100, 1, 1000, 10)$. These results illustrate the trade-off mentioned above. They make clear that more efficient deliveryman routes and more deliverymen can be used to reduce both the number and the distance traveled by vehicles if this is interesting from a cost perspective. However, when this is not the case, the vehicles are more intensively used to reduce deliverymen costs. From the first scenario to the last, the average number of deliverymen per vehicle drops from 2.59 to 1.93, which is a 25.48% decrease. Nevertheless, the average distance traveled by each vehicle and deliveryman does not change much from one scenario to the other since the fixed costs are much higher than the variable costs, enforcing that each vehicle and deliveryman is used as much as possible.

	Del < Veh	Del = Veh	Del > Veh
# of vehicles	6.22	6.28	7.56
# of deliverymen	16.11	15.33	14.56
First-level distance	423.00	439.39	522.83
Second-level distance	474.56	467.22	438.17

Table 6: Costs sensitivity analysis.

Furthermore, we look at other possibilities of cost reduction enabled by clever uses of multiple deliverymen in practice. Table 7 presents a base case with a limit of $M_L = 3$ deliverymen in each vehicle in which they travel at one-third of the vehicles' speed. This base case is compared to another with a limit of $M_L = 5$ deliverymen that travel at the same speed. Once again, these results prove that deliverymen can be used to reduce the number of vehicles used. Here, the number of vehicles is reduced by 17.85% and the first-level distance is reduced accordingly by 12.54%. Despite the increase in deliverymen costs, this leads to an overall cost reduction of 11.32%. Another comparison is made with a case of fast deliverymen (twice the vehicles' speed), which could represent a case with drones, bicycles, or motorcycles as deliverymen, instead of walking carriers. Even though the number of deliverymen in each vehicle remains $M_L = 3$, this increased speed allows for a great reduction on the service time in each cluster, leading to better first-level routes. The average number of vehicles is reduced by 22.35%, the number of deliverymen by 25.88%, and the vehicles distance by 17.89%, leading to a cost reduction of 20.52%.

A beneficial side effect of the business model incorporated by the VRPTWMD2R is a reduction on the emission of greenhouse gases (GHGs) and other pollutants. If the deliverymen are walking carriers, bicycles, or drones, for instance, GHGs emissions are much smaller in the second-level routes than in the first-level routes. Since the adoption of more deliverymen leads to reduced vehicle usage, as demonstrated above, this also reduces the environmental harm of the delivery. As presented in Table 7, the distance traveled by the vehicles can be reduced by more than 15% with the proper usage of deliverymen, creating a much greener last-mile delivery system while reducing operational costs.

In conclusion, the presented results demonstrate the importance of properly evaluating

	Base case $M_L = 3$ 1/3 of vehicles' speed	More deliverymen $M_L = 5$ 1/3 of vehicles' speed	Fast deliverymen $M_L = 3$ $2\times$ the vehicles' speed
# of vehicles	6.22	5.11	4.83
# of deliverymen	16.11	17.78	11.94
First-level distance	423.00	369.94	347.33
Second-level distance	474.56	530.33	464.06
Total cost	12,537.89	11,118.67	9,965.17

Table 7: Further advantages of multiple deliverymen.

the deliveryman routes and integrating them in a cost-effective manner with the vehicle routes. It has been shown that the adequate usage of deliverymen can reduce overall costs, the usage and number of vehicles, and the emission of GHGs and other pollutants. This creates the opportunity of devising less costly and greener operations.

7 Conclusion

In this work, we have introduced a novel problem in the literature called the vehicle routing problem with time windows, multiple deliverymen, and two-level routing. This problem is an extension of the vehicle routing problem with time windows and multiple deliverymen in which we incorporate the routes traveled by the deliverymen. We formally define and formulate the problem, propose valid inequalities for this formulation, and develop a branch-and-Benders-cut algorithm to solve it efficiently.

The results of computational experiments show the relevance of including more than one deliveryman in each vehicle and properly optimizing their routes inside the clusters. We have shown that this evaluation leads to a significant cost reduction and directly impacts the number of vehicles and their routes in the solution. The experiments confirmed the suitability of the proposed methodology. The proposed BBC solves 129 out of 145 instances to proven optimality, with an average processing time of less than 1,000s. The proposed method is capable of solving instances of realistic sizes. Moreover, we have performed a sensitivity analysis on the costs that highlighted opportunities to improve the usage of multiple deliverymen, such as increasing the number of deliverymen in each vehicle and adopting faster deliverymen (e.g., drones, bicycles, and motorcycles). We have also discussed beneficial environmental effects of this business model, which are relevant in urban logistics.

Finally, some possibilities of future work are extending the problem further and proposing other solution methods. Interesting extensions would be considering pickup-and-delivery schemes, heterogeneous fleet (especially in the first level), or uncertainties (e.g., in the demand or travel times). Regarding new methods, the development of heuristics and metaheuristics, or their combination with the proposed BBC to create hybrid methods, could lead to good solutions for even larger instances.

References

Alfandari, L., Ljubić, I., and De Melo da Silva, M. (2022). "A tailored Benders decomposition approach for last-mile delivery with autonomous robots". In: *European Journal of Operational Research* 299(2), pp. 510–525.

Álvarez, A. and Munari, P. (2016). "Metaheuristic approaches for the vehicle routing problem with time windows and multiple deliverymen". In: *Gestão e Produção* 23(2), pp. 279–293.

Álvarez, A. and Munari, P. (2017). "An exact hybrid method for the vehicle routing problem with time windows and multiple deliverymen". In: *Computers & Operations Research* 83, pp. 1–12.

Ascheuer, N., Fischetti, M., and Grötschel, M. (2001). "Solving the Asymmetric Travelling Salesman Problem with time windows by branch-and-cut". In: *Mathematical Programming* 90, pp. 475–506.

Bayliss, C., Bektaş, T., Tjon-Soei-Len, V., and Rohner, R. (2023). "Designing a multimodal and variable-echelon delivery system for last-mile logistics". In: *European Journal of Operational Research* 307(2), pp. 645–662.

Benders, J. F. (1962). "Partitioning procedures for solving mixed-variables programming problems". In: *Numerische Mathematik* 4(1), pp. 238–252.

Boysen, N., Fedtke, S., and Schwerdfeger, S. (2021). "Last-mile delivery concepts: a survey from an operational research perspective". In: *OR Spectrum* 43, pp. 1–58.

Codato, G. and Fischetti, M. (2006). "Combinatorial Benders' Cuts for Mixed-Integer Linear Programming". In: *Operations Research* 54(4), pp. 756–766.

Cormen, T. H., Leiserson, C. E., Rivest, R. L., and Stein, C. (2009). *Introduction to Algorithms*. 4th ed. The MIT Press: Boston, USA.

Hooker, J. N. and Ottosson, G. (2003). "Logic-based Benders decomposition". In: *Mathematical Programming* 96(1), pp. 33–60.

Le Colleter, T., Dumez, D., Lehuédé, F., and Péton, O. (2023). "Small and large neighborhood search for the park-and-loop routing problem with parking selection". In: *European Journal of Operational Research* 308(3), pp. 1233–1248.

Li, H., Chen, J., Wang, F., and Bai, M. (2021). "Ground-vehicle and unmanned-aerial-vehicle routing problems from two-echelon scheme perspective: A review". In: *European Journal of Operational Research* 294(3), pp. 1078–1095.

Martinez-Sykora, A., Lamas-Fernandez, C., Bektaş, T., Cherrett, T., and Allen, J. (2020). "Optimised solutions to the last-mile delivery problem in London using a combination of walking and driving". In: *Annals of Operations Research* 295, pp. 645–693.

Moreno, A., Munari, P., and Alem, D. (2019). "A branch-and-Benders-cut algorithm for the crew scheduling and routing problem in road restoration". In: *European Journal of Operational Research* 275(1), pp. 16–34.

Moreno, A., Munari, P., and Alem, D. (2020). "Decomposition-based algorithms for the crew scheduling and routing problem in road restoration". In: *Computers & Operations Research* 119, p. 104935.

Moshref-Javadi, M. and Winkenbach, M. (2021). "Applications and research avenues for drone-based models in logistics: A classification and review". In: *Expert Systems with Applications* 177, p. 114854.

Munari, P. and Morabito, R. (2018). "A branch-price-and-cut algorithm for the vehicle routing problem with time windows and multiple deliverymen". In: *TOP* 26(3), pp. 437–464.

Murray, C. C. and Chu, A. G. (2015). "The flying sidekick traveling salesman problem: Optimization of drone-assisted parcel delivery". In: *Transportation Research Part C: Emerging Technologies* 54, pp. 86–109.

Ostermeier, M., Heimfarth, A., and Hübner, A. (2023). "The multi-vehicle truck-and-robot routing problem for last-mile delivery". In: *European Journal of Operational Research* 310(2), pp. 680–697.

Ouyang, Z., Leung, E. K., and Huang, G. Q. (2023). "Community logistics and dynamic community partitioning: A new approach for solving e-commerce last mile delivery". In: *European Journal of Operational Research* 307(1), pp. 140–156.

Parada, L., Legault, R., Côté, J.-F., and Gendreau, M. (2023). *The disaggregated integer L-shaped method for the stochastic vehicle routing problem.* arXiv: `2212.06962 [math.OC]`.

Pureza, V., Morabito, R., and Reimann, M. (2012). "Vehicle routing with multiple deliverymen: Modeling and heuristic approaches for the VRPTW". In: *European Journal of Operational Research* 218(3), pp. 636–647.

Senarclens de Grancy, G. (2015). "An adaptive metaheuristic for vehicle routing problems with time windows and multiple service workers". In: *Journal of Universal Computer Science* 21(9), pp. 1143–1167.

Senarclens de Grancy, G. and Reimann, M. (2014). "Vehicle routing problems with time windows and multiple service workers: a systematic comparison between ACO and GRASP". In: *Central European Journal of Operations Research* 24(1), pp. 29–48.

Senarclens de Grancy, G. and Reimann, M. (2015). "Evaluating two new heuristics for constructing customer clusters in a VRPTW with multiple service workers". In: *Central European Journal of Operations Research* 23(2), pp. 479–500.

Solomon, M. M. (1987). "Algorithms for the Vehicle Routing and Scheduling Problems with Time Window Constraints". In: *Operations Research* 35(2), pp. 254–265.

Souza Neto, J. F. and Pureza, V. (2016). "Modeling and solving a rich vehicle routing problem for the delivery of goods in urban areas". In: *Pesquisa Operacional* 36(3), pp. 421–446.

Wehbi, L., Bektaş, T., and Iris, Ç. (2022). "Optimising vehicle and on-foot porter routing in urban logistics". In: *Transportation Research Part D: Transport and Environment* 109, p. 103371.