Centre interuniversitaire de recherche sur les réseaux d'entreprise, la logistique et le transport

Interuniversity Research Centre on Enterprise Networks, Logistics and Transportation

Alternative Formulations and Improved Bounds for the Multi-Depot Fleet Size and Mix Vehicle Routing

Rahma Lahyani
Leandro C. Coelho
Jacques Renaud

August 2015

CIRRELT-2015-36

Document de travail également publié par la Faculté des sciences de l'administration de l'Université Laval, sous le numéro FSA-2015-009.

Alternative Formulations and Improved Bounds for the Multi-Depot Fleet Size and Mix Vehicle Routing

Rahma Lahyani*, Leandro C. Coelho, Jacques Renaud

Interuniversity Research Centre on Enterprise Networks, Logistics and Transportation (CIRRELT) and Department of Operations and Decision Systems, 2325 de la Terrasse, Université Laval, Québec, Canada G1V 0A6

Abstract. In this paper we model and solve the multi-depot fleet size and mix vehicle routing problem (MDFSMVRP). This problem extends the multi-depot vehicle routing problem and the fleet size and mix vehicle routing problem, two logistics problems that have been extensively studied for many decades. This difficult transportation problem combines complex assignment and routing decisions under the objective of minimizing fixed vehicle costs and variable routing costs. We first propose five distinct formulations to model the MDFSMVRP. We introduce a three-index formulation with an explicit vehicle index and a compact two-index formulation in which only vehicle types are identified. Other formulations are obtained by defining aggregated and disaggregated loading variables. The last formulation makes use of capacity-indexed variables. For each formulation, we propose sets of known and new valid inequalities to strengthen them, including symmetry breaking, lexicographic ordering, routing and rounded capacity cuts, among others. We then implement branch-and-cut and branch-and bound algorithms for these formulations. We compare the bounds provided by the formulations on a commonly used set of instances in the MDFSMVRP literature, containing up to nine depots and 360 customers, and on newly generated instances. Our in-depth analysis of the five formulations shows which formulations tend to perform better on each type of instance. Moreover, we have considerably improved the lower bounds on all instances and significantly improved the quality of the upper bounds that can be obtained by means of currently available exact methods.

Keywords. Vehicle routing problem, multi-depot, heterogeneous fleet, formulation, mathematical model, exact solution.

Acknowledgements. This research was partly supported by grants 2014-05764 and 0172633 from the Natural Sciences and Engineering Research Council of Canada (NSERC). This support is gratefully acknowledged. We also thank Calcul Quebec for providing computing facilities.

---

* Corresponding author: Rahma.Lahyani@cirrelt.ca

Dépôt légal – Bibliothèque et Archives nationales du Québec
Bibliothèque et Archives Canada, 2015

1 Introduction

Distribution problems are central to the performance of many industries. The area of transportation has been widely studied, notably the vehicle routing problem (VRP) [42] which has attracted the interest of many researchers over more than 50 years [23] and is still among the most prominent and widely studied combinatorial optimization problems. Several different exact and heuristic algorithms have been proposed since the seminal paper of Dantzig et al. [10], and in the past decade a myriad of practical applications have emerged, describing many variants of the classical capacitated VRP [9]. These variants often incorporate ad hoc decisions or constraints to address challenging problems observed from practice and are referred to as *rich VRPs* [21]. They call for novel models and algorithms capable of solving new practical logistics problems.

In this paper we model and solve one of these variants, namely the multi-depot fleet size and mix vehicle routing problem (MDFSMVRP). This problem is a direct generalization of the classical VRP by considering multiple depots to serve a set of customers with known demands, and different types of vehicles. The problem combines three decisions simultaneously: selecting the number of vehicles of each type, planning vehicle routes and assigning routes to depots. Each vehicle is characterized by a fixed usage cost and a variable cost proportional to the traveled distance. The number of available vehicles of each type is assumed to be unlimited. The simultaneous optimization of the best fleet composition, the best vehicle routes and the depot choice substantiates the richness of this problem. The MDFSMVRP consists of designing a set of vehicle routes, each starting and ending at the same depot, visiting each customer exactly once, and respecting the capacity of the vehicles. The objective is to minimize the total fixed and variable routing costs.

The literature dealing with the MDFSMVRP is rather scarce. We are aware of three works focusing on this particular variant. A seminal work on the MDFSMVRP is due to Salhi and Sari [38]. The authors propose a multi-level composite heuristic based on integrating and modifying efficient heuristics designed for the single depot fleet size and mix vehicle routing problem (FSMVRP). Their method relies on switching to a more powerful and expensive neighborhood when moving to a superior level. The authors integrate reduction tests and refinement modules in the heuristic to speed up some of its steps. Seventeen years later, Salhi et al. [39] propose a mixed integer linear program to formulate the problem and a set of valid inequalities to tighten it. They also propose a variable neighborhood search metaheuristic. The method distinguishes between customers served from their nearest depots and borderline customers and makes use of local search heuristics and Dijkstra's algorithm to determine the optimal sequencing. They derive lower and upper bounds using a three-hour execution of CPLEX and provide percentage gaps computed using the best known bounds. Recently, Vidal et al. [43] propose a unified algorithmic framework tackling different classes of multi-depot VRPs with and without fleet mix. They introduce a bidirectional dynamic programming approach embedded in a multi-start iterated local search and a hybrid genetic search with advanced diversity control. The three published works assess the performance of their methods on the same testbed.

On the other hand, many books and book chapters have been devoted to study separately the two straightforward reductions of the MDFSMVRP. For focused and recent surveys for the FSMVRP we refer to Baldacci et al. [3], Irnich et al. [16], Koç et al. [19], and for the multi-depot VRP (MDVRP) to Montoya-Torres et al. [27]. More intricate and extended variants of this problem have also been studied. Mancini [26] and Rahimi-Vahed et al. [36] consider a closely related problem with multiple periods. Time related constraints have also received increased attention in the last few years, e.g., Bettinelli et al. [4], Xu and Jiang [44] and Koç et al. [18].

The MDFSMVRP is an NP-hard combinatorial problem since the VRP is NP-hard. Several authors explicitly outline the toughness of solving to optimality either the FSMVRP instances or the MDVRP instances, or even finding stronger bounds [33, 39].

Our contributions lie in adapting and proposing new formulations for the MDFSMVRP, as well as many valid inequalities. We compare five formulations against the one proposed by Salhi et al. [39]. Specifically, we propose a model based on a three-index VRP formulation introduced by Laporte and Nobert [24], to which we include new dimensions to account for each vehicle type. We then present a compact formulation derived from the two-index VRP model of Laporte [22], in which we create copies of the graph for each vehicle type, but we do not identify individual vehicles. Our third formulation is derived from the commodity flow model proposed in Salhi and Rand [37] and Yaman [45], which we modify to consider multiple depots. This formulation makes use of loading variables to model capacity and subtour elimination constraints. We obtain our fourth formulation by disaggregating the loading variables by vehicle type, as in Yaman [45]. Finally, the last formulation we propose is derived from the model of Pessoa et al. [33] for the FSMVRP, which is compact enough to enumerate all variables and constraints, and to which we incorporate new procedures to reduce the number of variables. We compare these five formulations in order to provide tighter bounds for this rich and difficult transportation problem. A subproduct of this research is to identify the origins and give credits to the main ideas used by our community to formulate many distribution problems. Thus, for each proposed formulation we provide the main references that put forward the modeling techniques and the valid inequalities that we use. This survey can greatly serve other researchers and students.

The remainder of this paper is organized as follows. In Section 2 we provide a formal description of the MDFSMVRP, followed by the presentation and introduction of the five mathematical models in Section 3. The algorithms used to solve these formulations are briefly presented in Section 4. The results of extensive computational experiments are presented in Section 5. Section 6 is devoted to our conclusions.

2 Problem description

The MDFSMVRP is formally defined on a directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{A})$, where $\mathcal{V}$ is the vertex set and $\mathcal{A}$ is the arc set. The vertex set $\mathcal{V}$ is partitioned into two subsets $\mathcal{V}_d = \{1, \ldots, m\}$ representing m depots, and $\mathcal{V}_c = \{m+1, \ldots, m+n\}$ representing n customers, such that $\mathcal{V} = \mathcal{V}_d \cup \mathcal{V}_c$. Each customer $i \in \mathcal{V}_c$ is associated with a non-negative demand q_i, while $q_i = 0$, $i \in \mathcal{V}_d$. The distance between nodes i and $j \in \mathcal{V}$ is represented by β_{ij}, thus the arc set $\mathcal{A}$ is composed of $\{(i, j) : i, j \in \mathcal{V}\}$. A set $\mathcal{K} = \{1, \ldots, K\}$ of heterogeneous vehicle types is available at each depot $d \in \mathcal{V}_d$. The fleet size is unlimited. For ease of notation, let a^k represents the number of vehicles of type k, bounded by $\underline{a^k} = 0$ and $\overline{a^k} = n$. We define a set $\mathcal{H} = \{1, \ldots, H\}$ including n copies of each vehicle type k, which are all available at each depot d, with $H = \overline{a^k}K$. Each vehicle type $k \in \mathcal{K}$ is associated with a capacity Q^k, a fixed cost F^k and a variable cost α^k per unit of distance.

A solution to the problem must determine routes that minimize the total costs such that each route must start and end at the same depot, each customer is visited exactly once, and the total demand of each route does not exceed the capacity of the selected vehicle.

3 Mathematical formulations

We now provide five different formulations for the MDFSMVRP. In Section 3.1 we present a model which explicitly considers all arcs, vehicles and depots. In Section 3.2 we show an adaptation of the compact two-index formulation, notably extending it to handle an heterogeneous fleet. Section 3.3 presents a commodity flow formulation in which capacity and subtour elimination constraints are expressed using flows. In Section 3.4 we introduce a model based on disaggregated loading variables by vehicle type. Finally, in Section 3.5 we present a capacity-indexed formulation for the problem at hand.

3.1 Explicit formulation

We first provide a three-index vehicle flow formulation for the symmetric case with an explicit vehicle index. The extension to an asymmetric version is straightforward. This model is based on the three-index vehicle flow formulation proposed by Laporte and Nobert [24] for the asymmetrical multi-depot VRP with homogeneous fleet, and on the model proposed by Toth and Vigo [41] for the single depot VRP. We define routing variables x_{ij}^{kd} equal to one if edge (i, j) is traversed by vehicle k housed at depot d, and equal to two for a round trip to customer j. Binary variables y_i^{kd} are equal to one if node i is visited by vehicle k from depot d. Note that in formulation F1 k refers to the vehicle index, not the vehicle type since all the available vehicles are explicitly considered. Note also that the set $\mathcal{A}$ contains only arcs with $i > j$, thus becoming an edge set, as required for this symmetric case. The problem can then be formulated as follows:

$$\text{(F1) minimize} \sum_{i\in\mathcal{V}_d}\sum_{k\in\mathcal{H}}\sum_{d\in\mathcal{V}_d} F^k y_i^{kd} + \sum_{i\in\mathcal{V}}\sum_{j\in\mathcal{V},i>j}\sum_{k\in\mathcal{H}}\sum_{d\in\mathcal{V}_d} \alpha^k \beta_{ij} x_{ij}^{kd} \tag{1}$$

subject to

$$\sum_{k\in\mathcal{H}}\sum_{d\in\mathcal{V}_d} y_i^{kd} = 1 \quad i \in \mathcal{V}_c \tag{2}$$

$$\sum_{j\in\mathcal{V},i>j} x_{ij}^{kd} + \sum_{j\in\mathcal{V},j>i} x_{ji}^{kd} = 2y_i^{kd} \quad i \in \mathcal{V}, k \in \mathcal{H}, d \in \mathcal{V}_d \tag{3}$$

$$y_i^{kd} \le y_d^{kd} \quad i \in \mathcal{V}_c, k \in \mathcal{H}, d \in \mathcal{V}_d \tag{4}$$

$$y_d^{kd} \le \sum_{i\in\mathcal{V}}\sum_{j\in\mathcal{V},i>j} x_{ij}^{kd} \quad k \in \mathcal{H}, d \in \mathcal{V}_d \tag{5}$$

$$2y_d^{kd} \le \sum_{i\in\mathcal{V}_c} x_{id}^{kd} \quad k \in \mathcal{H}, d \in \mathcal{V}_d \tag{6}$$

$$\sum_{i\in\mathcal{Z}}\sum_{j\in\mathcal{Z},i>j} x_{ij}^{kd} \le \sum_{i\in\mathcal{Z}} y_i^{kd} - y_z^{kd} \quad \mathcal{Z} \subseteq \mathcal{V}_c, |\mathcal{Z}| \ge 2, z \in \mathcal{Z}, k \in \mathcal{H}, d \in \mathcal{V}_d \tag{7}$$

$$\sum_{i\in\mathcal{V}_c} q_i y_i^{kd} \le Q^k \quad k \in \mathcal{H}, d \in \mathcal{V}_d \tag{8}$$

$$x_{ij}^{kd} = 0 \quad i \in \mathcal{V}_c, j \in \mathcal{V}_d, j \ne d, k \in \mathcal{H}, d \in \mathcal{V}_d \tag{9}$$

$$x_{ij}^{kd} \in \{0,1\} \quad i,j \in \mathcal{V}_c, i > j, k \in \mathcal{H}, d \in \mathcal{V}_d \tag{10}$$

$$x_{ij}^{kd} \in \{0,1,2\} \quad i \in \mathcal{V}_c, j \in \mathcal{V}_d, k \in \mathcal{H}, d \in \mathcal{V}_d \tag{11}$$

$$y_i^{kd} \in \{0,1\} \quad i \in \mathcal{V}, k \in \mathcal{H}, d \in \mathcal{V}_d. \tag{12}$$

The objective function (1) minimizes the total cost composed of fixed vehicle costs and variable routing costs. Constraints (2) impose that all customers must be visited exactly once. Constraints (3) are degree constraints and constraints (4) impose that if a customer is served by vehicle k housed at depot d, then vehicle k must leave the depot. Constraints (5) and (6) link the two types of variables of the problem. They ensure that if a vehicle k of depot d is used, then at least one customer i must be visited by this vehicle. Constraints (7) forbid subtours. Constraints (8) impose vehicle capacities, while constraints (9) remove some infeasible variables from the problem, namely ensuring that the vehicles leave and return to the same depot. The domain of the variables is enforced by constraints (10)–(12). This formulation has $nmH(1+m+\frac{n}{2}+\frac{m}{n})$ binary variables, $n+mH(3+2n+m+n(m-1))$ linear constraints and $O(2^n)$ subtour elimination constraints whose number grows exponentially with n. This is a large formulation which strongly depends on the number of available vehicles.

Model F1 is sufficient to represent the MDFSMVRP, however we can add some valid inequalities and lift some constraints to strengthen it. Equalities (13) remove unnecessary variables from the problem by forbidding trips between depots. Constraints (14) enforce restrictions related to the vehicle use. Specifically, each vehicle k housed at depot d is allowed to perform at most one trip.

$$x_{ij}^{kd} = 0 \quad i,j \in \mathcal{V}_d, i > j, k \in \mathcal{H}, d \in \mathcal{V}_d \tag{13}$$

$$\sum_{j \in \mathcal{V}_c} x_{jd}^{kd} \leq 2 \quad k \in \mathcal{H}, d \in \mathcal{V}_d. \tag{14}$$

To avoid symmetries due to the presence of identical vehicles at each depot, we introduce vehicle symmetry breaking constraints. Observe that (15) and (16) are only valid if the fleet is homogeneous. We define the set $\mathcal{H}^t \subset \mathcal{H}$ containing only the homogeneous vehicles of type t. Thus, constraints (15) state that vehicle k can only be dispatched if vehicle $k-1$ is already dispatched. Constraints (16) rank identical vehicles according to the index of the customers visited. These constraints are defined for each depot. They are inspired by those presented in Adulyasak et al. [1], Coelho and Laporte [7, 8] and Lahyani et al. [20].

$$y_d^{kd} \leq y_d^{k-1,d} \quad k \in \mathcal{H}^t \backslash \{\mathcal{H}_1^t\}, \mathcal{H}^t \subset \mathcal{H}, t \in \mathcal{K}, d \in \mathcal{V}_d \tag{15}$$

$$y_i^{kd} \leq \sum_{j \in \mathcal{V}_c, j<i} \sum_{h \in \mathcal{V}_d} y_j^{k-1,h} \quad i \in \mathcal{V}_c, k \in \mathcal{H}^t \backslash \{\mathcal{H}_1^t\}, \mathcal{H}^t \subset \mathcal{H}, t \in \mathcal{K}, d \in \mathcal{V}_d, \tag{16}$$

where $\mathcal{H}_1^t$ represents the first element of $\mathcal{H}^t$.

We also introduce a set of logical inequalities that enforce the relationships between routing and visiting variables. They are defined as follows:

$$y_d^{kd} \leq \sum_{i \in \mathcal{V}_c} y_i^{kd} \quad k \in \mathcal{H}, d \in \mathcal{V}_d \tag{17}$$

$$x_{id}^{kd} \leq 2y_i^{kd} \quad i \in \mathcal{V}_c, k \in \mathcal{H}, d \in \mathcal{V}_d \tag{18}$$

$$x_{ij}^{kd} \leq y_j^{kd} \quad i, j \in \mathcal{V}_c, i > j, k \in \mathcal{H}, d \in \mathcal{V}_d \tag{19}$$

$$\sum_{j \in \mathcal{V}_c} y_j^{kd} \leq \sum_{i \in \mathcal{V}} \sum_{j \in \mathcal{V}, i>j} x_{ij}^{kd} \quad k \in \mathcal{H}, d \in \mathcal{V}_d \tag{20}$$

$$2y_j^{kd} \leq \sum_{i \in \mathcal{V}_c} x_{id}^{kd} \quad j \in \mathcal{V}_c, k \in \mathcal{H}, d \in \mathcal{V}_d \tag{21}$$

$$\sum_{j \in \mathcal{V}, i>j} \sum_{k \in \mathcal{H}} \sum_{d \in \mathcal{V}_d} x_{ij}^{kd} + \sum_{j \in \mathcal{V}, i<j} \sum_{k \in \mathcal{H}} \sum_{d \in \mathcal{V}_d} x_{ji}^{kd} = 2 \quad i \in \mathcal{V}_c \tag{22}$$

$$\left\lceil \frac{\sum_{i \in \mathcal{V}_c} q_i}{max\{Q^k\}} \right\rceil \leq \sum_{i \in \mathcal{V}_c} \sum_{k \in \mathcal{H}} \sum_{d \in \mathcal{V}_d} x_{id}^{kd}. \tag{23}$$

Constraints (17)–(22) are referred to as routing cuts. The first ones replace the right hand side of constraints (5) by enforcing that at least one customer must be visited by vehicle k of depot d if this vehicle is used. We also note that constraints (17) are the sum over the customers in inequalities (4). Constraints (18) remove all edges (i, d) if customer i is not visited by vehicle k of depot d. Constraints (19) further remove variables by forbidding the use of edge (i, j) if customer j is not visited by vehicle k housed at depot d. Constraints (20) impose that the sum of customers visited by vehicle k is less than or equal to the sum of edges traversed by vehicle k. Constraints (21) impose the condition that if vehicle k of depot d is not used, then customer j cannot be visited by this vehicle. Equations (22) further define the degree constraints by imposing that each customer is visited once. Finally, constraints (23) are referred to as rounded capacity cuts [28, 33]. They impose a lower bound on the number of used vehicles. However, in the case it is not necessary to use the vehicle with the biggest capacity in one trip and if there is a considerable difference between $max\{Q^k\}$ and the capacity of the used vehicle, then the left hand side of constraints (23) may give a poor lower bound.

Constraints (24) and (25) are lexicographic ordering constraints. They are inspired from the ones defined in Sherali and Smith [40] and Adulyasak et al. [1]. Given the large coefficients that arise when dealing with large instances, these constraints are only added for small and medium size instances containing up to 60 customers.

$$\sum_{i=m+1}^{j} 2^{(j-i)} y_i^{kd} \leq \sum_{i=m+1}^{j} 2^{(j-i)} y_i^{k-1,d} \quad j \in \mathcal{V}_c, k \in \mathcal{H}^t \backslash \{\mathcal{H}_1^t\}, \mathcal{H}^t \subset \mathcal{H}, t \in \mathcal{K}, d \in \mathcal{V}_d \tag{24}$$

$$\sum_{i \in \mathcal{V}_c} 2^{(m+n-i)} y_i^{kd} \leq \sum_{i \in \mathcal{V}_c} 2^{(m+n-i)} y_i^{k-1,d} \quad k \in \mathcal{H}^t \backslash \{\mathcal{H}_1^t\}, \mathcal{H}^t \subset \mathcal{H}, t \in \mathcal{K}, d \in \mathcal{V}_d. \tag{25}$$

3.2 Compact formulation with implicit vehicle index

Formulation F1 has the drawback that the number of variables and constraints increases when the number of vehicle variables increases. These variables are linearly dependent on the number of customers in the instance because $\overline{a^k} = n$.

We now propose a formulation with implicit vehicle assignment as proposed in Laporte [22], Toth and Vigo [41] for the single depot VRP.

This formulation uses the same type of variables defined in Section 3.1, however the index k now (and for the remainder of this paper) refers to vehicle types instead of individual vehicles. This has the advantage of having one type of variable per vehicle type, instead of creating one variable per vehicle of each type. For the sake of briefness, we do not restate the whole definition of the variables, and refer to the ones already defined when the interpretation is straightforward. This compact formulation with implicit vehicle index can then be defined as follows:

$$\text{(F2) minimize} \sum_{j\in\mathcal{V}_c}\sum_{k\in\mathcal{K}}\sum_{d\in\mathcal{V}_d} 0.5F^k x_{jd}^{kd} + \sum_{i\in\mathcal{V}}\sum_{j\in\mathcal{V},i>j}\sum_{k\in\mathcal{K}}\sum_{d\in\mathcal{V}_d} \alpha^k\beta_{ij}x_{ij}^{kd} \tag{26}$$

subject to (2)–(6), (9)–(12) and to

$$\sum_{i\in\mathcal{S}}\sum_{j\in\mathcal{S},i>j} x_{ij}^{kd} \leq |S| - r(S) \quad \mathcal{S}\subseteq\mathcal{V}_c, \mathcal{S}\neq\emptyset, k\in\mathcal{K}, d\in\mathcal{V}_d. \tag{27}$$

When using a compact variables definition, the objective function (26) must be expressed by the variables x_{ij}^{kd}. Constraints (27) simultaneously replace constraints (7) and (8). They correspond to generalized subtour elimination constraints, and prevent capacity violation on each vehicle. This formulation has $nmK(1 + m + \frac{n}{2} + \frac{m}{n})$ binary variables, $n + mK(2 + 2n + m + n(m - 1))$ linear constraints and a number of linear subtour elimination constraints growing exponentially with n. F2 is much more compact than F1 since it depends on the number of vehicle types.

Several of the valid inequalities previously defined remain valid, namely (17)–(23). Alternatively, we can reinforce subtour elimination (27) by introducing inequalities (7). Constraints (7) are known to be efficient when solving the problem with a branch-and-cut algorithm. Both families of constraints (7) and (27) have a cardinality growing exponentially with n.

Vehicle symmetry breaking constraints and lexicographic ordering constraints no longer hold for this compact formulation because they require distinguishing between vehicle index and not vehicle types.

3.3 Compact formulation with loading variables

A main disadvantage of model F2 presented in Section 3.2 is that capacity constraints are not explicitly defined, requiring cuts to be added dynamically. This might lead to weak bounds at the early stages of its optimization. To overcome this situation, formulation F3 proposed in this section makes use of stronger constraints to handle capacity restrictions. We define additional continuous variables to help control the load of the vehicles. This model is based on the commodity flow formulation proposed by Garvin et al. [11] for an oil delivery problem and later extended by Gavish and Graves [12] to VRP variants. A similar formulation for the single depot VRP is given in Toth and Vigo [41]. Later, Baldacci et al. [3] extended this formulation for the VRP with heterogeneous fleet, Salhi and Rand [37] and Yaman [45] extended it for the FSMVRP, Salhi et al. [39] modified it to handle a VRP with multiple depots, and Koç et al. [18] amended it for the fleet size and mix location-routing problem with time windows. The formulation proposed in this section is quite different from the one proposed in Salhi et al. [39] for the MDFSMVRP as we define new routing variables y_i^{kd} in addition to x_{ij}^{kd}. Indeed, Bosch and Trick [5] highlight that adding variables and/or constraints to a formulation may strengthen the linear relaxation and provide improved formulations. They also state that for many problems, the use of integer variables, even when it is not required, may expand the capability of the model and help find an optimal solution.

The formulation is derived using the same four-index binary variables x_{ij}^{kd} and the visiting binary variables y_i^{kd} defined in Section 3.1 on a directed graph. We define new continuous variables z_{ij} representing the remaining load on the vehicle when traversing arc (i, j), i.e., after visiting node i and before visiting node j. Note that the loading variables could be defined only for the asymmetric version of the problem since the complete graph is considered. In what follows, we restate all the constraints of the problem dealing with routing variables x_{ij}^{kd}, since they are expressed differently from the constraints defined in models F1 and F2, despite having the same role. The formulation is defined by:

$$\text{(F3) minimize} \sum_{i\in\mathcal{V}_c}\sum_{k\in\mathcal{K}}\sum_{d\in\mathcal{V}_d} F^k x_{di}^{kd} + \sum_{i\in\mathcal{V}}\sum_{j\in\mathcal{V}}\sum_{k\in\mathcal{K}}\sum_{d\in\mathcal{V}_d} \alpha^k \beta_{ij} x_{ij}^{kd} \tag{28}$$

subject to (2), (4) and to:

$$\sum_{j\in\mathcal{V}} x_{ij}^{kd} + \sum_{j\in\mathcal{V}} x_{ji}^{kd} = 2y_i^{kd} \quad i \in \mathcal{V}_c, k \in \mathcal{K}, d \in \mathcal{V}_d \tag{29}$$

$$\sum_{i \in \mathcal{V}} x_{ij}^{kd} = \sum_{i \in \mathcal{V}} x_{ji}^{kd} \quad j \in \mathcal{V}, k \in \mathcal{K}, d \in \mathcal{V}_d \tag{30}$$

$$y_d^{kd} \leq \sum_{i \in \mathcal{V}} \sum_{j \in \mathcal{V}} x_{ij}^{kd} \quad k \in \mathcal{K}, d \in \mathcal{V}_d \tag{31}$$

$$2y_d^{kd} \leq \sum_{j \in \mathcal{V}_c} x_{jd}^{kd} + \sum_{j \in \mathcal{V}_c} x_{dj}^{kd} \quad k \in \mathcal{K}, d \in \mathcal{V}_d \tag{32}$$

$$x_{ij}^{kd} = 0 \quad i \in \mathcal{V}_c, j \in \mathcal{V}_d, j \neq d, k \in \mathcal{K}, d \in \mathcal{V}_d \tag{33}$$

$$x_{ij}^{kd} = 0 \quad i \in \mathcal{V}_d, i \neq d, j \in \mathcal{V}_c, k \in \mathcal{K}, d \in \mathcal{V}_d \tag{34}$$

$$\sum_{i \in \mathcal{V}} z_{ij} - \sum_{i \in \mathcal{V}} z_{ij} = q_j \quad j \in \mathcal{V}_c \tag{35}$$

$$\sum_{i \in \mathcal{V}_d} \sum_{j \in \mathcal{V}_c} z_{ij} = \sum_{j \in \mathcal{V}_c} q_j \tag{36}$$

$$z_{ij} \leq \sum_{k \in \mathcal{K}} \sum_{d \in \mathcal{V}_d} (Q_k - q_i) x_{ij}^{kd} \quad i \in \mathcal{V}, j \in \mathcal{V}_c \tag{37}$$

$$x_{ij}^{kd} \in \{0, 1\} \quad i, j \in \mathcal{V}, k \in \mathcal{K}, d \in \mathcal{V}_d \tag{38}$$

$$y_i^{kd} \in \{0, 1\} \quad i \in \mathcal{V}, k \in \mathcal{K}, d \in \mathcal{V}_d \tag{39}$$

$$z_{ij} \geq 0 \quad i, j \in \mathcal{V}. \tag{40}$$

The objective function (28) minimizes the total routing costs. Equations (2) enforce that each customer must be visited exactly once. Constraints (29) and (30) replace the flow conservation constraints (3) defined in model F1. Constraints (4), (31) and (32) are equivalent to constraints (4)–(6) in model F1. They enforce that only used vehicles may serve customers. Similarly, constraints (33) and (34) are equivalent to constraints (9). They guarantee that a vehicle leaves and returns to the same depot. Constraints (35)–(37) are specific to the commodity flow formulation. They impose both the connectivity of the solution and the vehicle capacity constraints. In particular, constraints (35) guarantee that each customer demand is satisfied. Summing up these constraints yields constraint (36) which states that the total load leaving all depots must be equal to the total customers demands. Constraints (37) bound the load on each arc (i, j), i.e., after visiting node i the load on arc (i, j) plus the demand of node i cannot exceed the capacity of the vehicle used.

Constraints (38)–(40) define the domain and nature of the variables. Formulation F3 has $mK(|\mathcal{A}|+n+m)$ binary variables, $|\mathcal{A}|$ continuous variables and $1+n(2+n+m)+mK(2+3n+m+2n(m-1))$ constraints. It has the advantage that the connectivity constraints are initially polynomial in size, unlike models F1 and F2 which require a branch-and-cut algorithm to dynamically add subtour elimination constraints which are exponential in number.

Because of the way new variables z_{ij} are defined, it is possible to further tighten this formulation. We introduce bounding constraints and we remove unnecessary variables from the problem, as in Salhi et al. [39]. Constraints (41) impose a lower bound on loading variables. They state that the total load of arc (i, j) must be at least equal to the demand of node i. We fix some variables to zero in equalities (42)–(46). Constraints (42) impose that a vehicle returns to the depot empty and constraints (43) and (44) forbid carrying a load between depots or between a customer and itself. Constraints (45) and (46) remove arcs between depots and between a customer and itself.

$$z_{ij} \geq \sum_{k\in\mathcal{K}} \sum_{d\in\mathcal{V}_d} q_j x_{ij}^{kd} \quad i \in \mathcal{V}_c, j \in \mathcal{V}_c \tag{41}$$

$$z_{ij} = 0 \quad i \in \mathcal{V}_c, j \in \mathcal{V}_d \tag{42}$$

$$z_{ij} = 0 \quad i, j \in \mathcal{V}_d \tag{43}$$

$$z_{ii} = 0 \quad i \in \mathcal{V}_c \tag{44}$$

$$x_{ij}^{kd} = 0 \quad i, j \in \mathcal{V}_d, k \in \mathcal{K}, d \in \mathcal{V}_d \tag{45}$$

$$x_{ii}^{kd} = 0 \quad i \in \mathcal{V}, k \in \mathcal{K}, d \in \mathcal{V}_d. \tag{46}$$

Constraints (47) and (48) enhance the flow conservation of the problem by imposing that the total flow entering a node must equal to the total flow leaving the node. Karaoglan et al. [17] have introduced several classes of valid inequalities for the location-routing problem with simultaneous pick-up and delivery. Some of these constraints have been extended to the fleet size and mix location-routing problem with time windows in Koç et al. [18]. We adapt these constraints in (49)–(51) to the MDFSMVRP. They exclude illegal vehicle routes that do not start and end at the same depot. Constraints (52) represent a special case of subtour elimination constraints on 2-node sets. Constraints (53) bound the number of vehicles trips.

$$\sum_{i \in \mathcal{V}} \sum_{k \in \mathcal{K}} \sum_{d \in \mathcal{V}_d} x_{ij}^{kd} = 1 \quad j \in \mathcal{V}_c \tag{47}$$

$$\sum_{j \in \mathcal{V}} \sum_{k \in \mathcal{K}} \sum_{d \in \mathcal{V}_d} x_{ij}^{kd} = 1 \quad i \in \mathcal{V}_c \tag{48}$$

$$\sum_{k \in \mathcal{K}} x_{id}^{kd} \leq \sum_{k \in \mathcal{K}} y_i^{kd} \quad i \in \mathcal{V}_c, d \in \mathcal{V}_d \tag{49}$$

$$\sum_{k \in \mathcal{K}} x_{di}^{kd} \leq \sum_{k \in \mathcal{K}} y_i^{kd} \quad i \in \mathcal{V}_c, d \in \mathcal{V}_d \tag{50}$$

$$\sum_{k \in \mathcal{K}} x_{ij}^{kd} + \sum_{k \in \mathcal{K}} y_i^{kd} + \sum_{k \in \mathcal{K}} \sum_{h \in \mathcal{V}_d, h \neq d} y_j^{kh} \leq 2 \quad i, j \in \mathcal{V}_c, i \neq j, d \in \mathcal{V}_d \tag{51}$$

$$x_{ij}^{kd} + x_{ji}^{kd} \leq 1 \quad i, j \in \mathcal{V}_c, k \in \mathcal{K}, d \in \mathcal{V}_d \tag{52}$$

$$\left\lceil \frac{\sum_{i \in \mathcal{V}_c} q_i}{max\{Q^k\}} \right\rceil \leq \sum_{i \in \mathcal{V}_c} \sum_{k \in \mathcal{K}} \sum_{d \in \mathcal{V}_d} x_{di}^{kd}. \tag{53}$$

3.4 Compact formulation with disaggregated loading variables

In this section, we propose a more detailed formulation based on F3 for the MDFSMVRP, referred to as F4. The motivation is to carry information related to the vehicle type on each arc by disaggregating the loading variables z_{ij}. We define new continuous variables z_{ij}^k, such that $z_{ij} = \sum_{k \in \mathcal{K}} z_{ij}^k$. This model is inspired from the work of Yaman [45] for the FSMVRP. The model is defined by minimizing (28) subject to (2), (4), (29)–(34), (38), (39) and to:

$$\sum_{i \in \mathcal{V}_d} \sum_{j \in \mathcal{V}_c} \sum_{k \in \mathcal{K}} z_{ij}^k = \sum_{j \in \mathcal{V}_c} q_j \tag{54}$$

$$\sum_{i \in \mathcal{V}} z_{ij}^k - \sum_{i \in \mathcal{V}} z_{ij}^k = \sum_{d \in \mathcal{V}_d} q_j y_j^{kd} \quad j \in \mathcal{V}_c, k \in \mathcal{K} \tag{55}$$

$$z_{ij}^k \leq \sum_{d \in \mathcal{V}_d} (Q_k - q_i) x_{ij}^{kd} \quad i \in \mathcal{V}, j \in \mathcal{V}_c, k \in \mathcal{K} \tag{56}$$

$$z_{ij}^k \geq 0 \quad i, j \in \mathcal{V}, k \in \mathcal{K}. \tag{57}$$

Constraints (54)–(56) have a similar meaning as constraints (35)–(37) of model F3. The only exception is that they provide more precision on the vehicle type carrying the load on arc (i, j). Formulation F4 has a few more variables and constraints compared to F3. It contains $(n + n(n + m))(K - 1)$ more constraints due to constraints (55)–(56) and $|\mathcal{A}|(K - 1)$ more continuous variables. If $K < n$, both formulations have $O(n^2m^2)$ constraints. However, F4 has $O(n|\mathcal{A}|)$ continuous variables while F3 has $O(|\mathcal{A}|)$.

Model F4 can also be strengthened by (45)–(53), while constraints (41)–(44) must be replaced by:

$$z_{ij}^k \geq \sum_{d \in \mathcal{V}_d} q_j x_{ij}^{kd} \quad i \in \mathcal{V}_c, j \in \mathcal{V}_c, k \in \mathcal{K} \tag{58}$$

$$z_{ij}^k = 0 \quad i \in \mathcal{V}_c, j \in \mathcal{V}_d, k \in \mathcal{K} \tag{59}$$

$$z_{ij}^k = 0 \quad i, j \in \mathcal{V}_d, k \in \mathcal{K} \tag{60}$$

$$z_{ii}^k = 0 \quad i \in \mathcal{V}_c, k \in \mathcal{K}. \tag{61}$$

3.5 Capacity-indexed formulation

In this section, we propose a novel formulation to model VRPs, referred to as capacity-indexed formulation. This type of formulation has only appeared a few times for basic variants of VRPs. A seminal paper proposing a capacity-indexed formulation for the time-dependent traveling salesman problem is due to Picard and Queyranne [34]. Godinho et al. [15] used it for the case of unitary demands. Later, Pessoa et al. [32] and Poggi de Aragão and Uchoa [35] propose a similar formulation for the asymmetric VRP, and Pessoa et al. [31] and Pessoa et al. [33] extend this model to handle the asymmetric VRP with heterogeneous fleet.

We define new binary variables x_{ij}^{kdq} equal to one if and only if vehicle type k housed at depot d traverses arc (i, j) with a load of q units. This variable indicates the current load of a given vehicle type housed at a given depot on a given arc, unlike the commodity flow formulations (F3 and F4) that require the definition of continuous variables to convey similar information. This model can then be formulated as follows:

$$\text{(F5) minimize} \sum_{i \in \mathcal{V}_c} \sum_{k \in \mathcal{K}} \sum_{d \in \mathcal{V}_d} \sum_{q=1}^{Q^k} F^k x_{di}^{kdq} + \sum_{i \in \mathcal{V}} \sum_{j \in \mathcal{V}} \sum_{k \in \mathcal{K}} \sum_{d \in \mathcal{V}_d} \sum_{q=0}^{Q^k} \alpha^k \beta_{ij} x_{ij}^{kdq} \tag{62}$$

subject to

$$\sum_{j \in \mathcal{V}} \sum_{k \in \mathcal{K}} \sum_{d \in \mathcal{V}_d} \sum_{q=1}^{Q^k} x_{ji}^{kdq} = 1 \quad i \in \mathcal{V}_c \tag{63}$$

$$\sum_{i \in \mathcal{V}_c} \sum_{q=1}^{Q^k} x_{di}^{kdq} = \sum_{i \in \mathcal{V}_c} x_{id}^{kd0} \quad k \in \mathcal{K}, d \in \mathcal{V}_d \tag{64}$$

$$\sum_{j \in \mathcal{V}} x_{ji}^{kdq} = \sum_{j \in \mathcal{V}} x_{ij}^{kd(q-q_i)} \quad i \in \mathcal{V}_c, k \in \mathcal{K}, d \in \mathcal{V}_d, q = \{q_i, \ldots, Q^k\} \tag{65}$$

$$x_{ij}^{kdq} = 0 \quad i \in \mathcal{V}_c, j \in \mathcal{V}_d, k \in \mathcal{K}, d \in \mathcal{V}_d, q = \{1, \ldots, Q^k\} \tag{66}$$

$$x_{ij}^{kdq} = 0 \quad i \in \mathcal{V}, j \in \mathcal{V}_c, k \in \mathcal{K}, d \in \mathcal{V}_d, q = \{0, \ldots, q_j - 1\} \tag{67}$$

$$x_{ij}^{kdq} = 0 \quad i \in \mathcal{V}_c, j \in \mathcal{V}_d, j \neq d, k \in \mathcal{K}, d \in \mathcal{V}_d, q = \{0, \ldots, Q^k\} \tag{68}$$

$$x_{ij}^{kdq} = 0 \quad i \in \mathcal{V}_d, i \neq d, j \in \mathcal{V}_c, k \in \mathcal{K}, d \in \mathcal{V}_d, q = \{0, \ldots, Q^k\} \tag{69}$$

$$x_{ij}^{kdq} \in \{0, 1\} \quad i, j \in \mathcal{V}, k \in \mathcal{K}, d \in \mathcal{V}_d, q = \{0, \ldots, Q^k\}. \tag{70}$$

The total routing costs are minimized in (62). Equations (63) are in-degree constraints. They ensure that each customer is visited exactly once. Constraints (64) ensure flow conservation and guarantee that if a vehicle of type k leaves a depot d with a load q then it must return to this depot with a load equal to 0. The connectivity of the solution and the vehicle capacity requirements are ensured due to constraints (65). If vehicle k carrying a load $q_i \leq q \leq Q^k$ enters a node i, then it must leave it with a load equal to $q - q_i$. Infeasible and unnecessary variables are removed with equalities (66)–(69). Constraints (66) forbid vehicles to return to the depot with a load different from zero. Constraints (67) state that a vehicle k traversing an arc (i, j) must not carry a load q lower than the demand of node j. Constraints (68) and (69) are equivalent to constraints (9) in models F1 and F2 and to constraints (33) and (34) in models F3 and F4. They ensure that a vehicle route must start and end at the same depot. Constraints (70) define the domain of the capacity-indexed variables. Formulation F5 has $m|\mathcal{A}|(\sum_{k=1}^{K}(Q^k+1))$ binary variables and $n + mK(1 + \sum_{i\in\mathcal{V}_c}(Q^k - q_i + 1) + (n+m)\sum_{i\in\mathcal{V}_c} q_i) + nm(m\sum_{k=1}^{K} Q^k + 2(m-1)\sum_{k=1}^{K}(Q^k+1))$ linear constraints.

In order to reduce the research space when using capacity-indexed variables, one can further remove unnecessary variables. We eliminate variables related to vehicle k traversing an arc (i,j) with an irrelevant load, i.e., after visiting a node i the vehicle should not carry a load between $Q^k - q_i$ and the capacity of vehicle k, Q^k. Those unnecessary variables are removed with equalities (71). We also remove infeasible arcs as in the previous models, through equations (72) and (73).

$$x_{ij}^{kdq} = 0 \quad i \in \mathcal{V}, j \in \mathcal{V}_c, k \in \mathcal{K}, d \in \mathcal{V}_d, Q^k - q_i < q \leq Q^k \tag{71}$$

$$x_{ii}^{kdq} = 0 \quad i \in \mathcal{V}, k \in \mathcal{K}, d \in \mathcal{V}_d, q = \{0, \ldots, Q^k\} \tag{72}$$

$$x_{ij}^{kdq} = 0 \quad i, j \in \mathcal{V}_d, k \in \mathcal{K}, d \in \mathcal{V}_d, q = \{0, \ldots, Q^k\}. \tag{73}$$

Solving the MDFSMVRP directly with this formulation is practical only for small values of Q^k. We derive new valid inequalities in the form of balance and capacity constraints and routing constraints, which impose bounds on the binary variables. Constraints (74) and (75) are inspired from those proposed in Pessoa et al. [33]. They impose a lower bound on the number of vehicle routes and the number of variables, respectively. Equations (76) are balance constraints. They state that if vehicle k traversing arc (i,j) enters node i then the load delivered to node i must be exactly q_i.

$$\left\lceil \frac{\sum_{i\in\mathcal{V}_c} q_i}{max\{Q^k\}} \right\rceil \leq \sum_{i\in\mathcal{V}_c}\sum_{k\in\mathcal{K}}\sum_{d\in\mathcal{V}_d}\sum_{q=q_i}^{Q^k} x_{di}^{kdq} \tag{74}$$

$$\sum_{i\in\mathcal{V}_c} q_i \leq \sum_{i\in\mathcal{V}}\sum_{j\in\mathcal{V}_c}\sum_{k\in\mathcal{K}}\sum_{d\in\mathcal{V}_d}\sum_{q=q_i}^{Q^k} q x_{ij}^{kdq} \tag{75}$$

$$\sum_{j\in\mathcal{V}}\sum_{k\in\mathcal{K}}\sum_{d\in\mathcal{V}_d}\sum_{q=q_i}^{Q^k-q_j} q x_{ji}^{kdq} - \sum_{j\in\mathcal{V}}\sum_{k\in\mathcal{K}}\sum_{d\in\mathcal{V}_d}\sum_{q=q_j}^{Q^k-q_i} q x_{ij}^{kdq} = q_i \quad i \in \mathcal{V}_c. \tag{76}$$

Constraints (77)–(79) are referred to as routing constraints, as a way to ensure that if there is an arc (i, j) related to vehicle k and linking two customers i and j, i.e., (77) holds, then there must be at least one arc traversed by k and returning to depot d, i.e., (78) holds. Equalities (79) are outgoing edges, they reinforce equations (63).

$$\sum_{q=q_j}^{Q^k-q_i} x_{ij}^{kdq} \leq \sum_{h\in\mathcal{V}_c} x_{hd}^{kd0} \quad i,j \in \mathcal{V}_c, k \in \mathcal{K}, d \in \mathcal{V}_d \tag{77}$$

$$x_{hd}^{kd0} \leq \sum_{i\in\mathcal{V}} \sum_{j\in\mathcal{V}_c} \sum_{q=q_j}^{Q^k-q_i} x_{ij}^{kdq} \quad h \in \mathcal{V}_c, k \in \mathcal{K}, d \in \mathcal{V}_d \tag{78}$$

$$\sum_{j\in\mathcal{V}} \sum_{k\in\mathcal{K}} \sum_{d\in\mathcal{V}_d} \sum_{q=0}^{Q^k} x_{ij}^{kdq} = 1 \quad i \in \mathcal{V}_c. \tag{79}$$

4 Solution algorithms

The formulations presented in Sections 3.3, 3.4 and 3.5 can be explicitly generated and one can enumerate all its variables and constraints. These can then be fed into a general purpose solver and solutions are obtained by branch-and-bound. However, the models presented in Sections 3.1 and 3.2 cannot be fully generated due to constraints (7) and (27) which are in the order of $O(2^n)$. Thus, one needs to dynamically generate them only if they are found to be violated in a partial solution. The exact algorithm we present is then a classical branch-and-cut which works as follows. At a generic node of the search tree, a linear program including a subset of the subtour elimination constraints is solved, a search for violated constraints is performed, appropriate valid inequalities are added to eliminate subtours, and the current subproblem is then reoptimized. This process is reiterated until a feasible or dominated solution is reached, or until no more cuts can be added. At this point, branching on a fractional variable occurs. We provide a sketch of the branch-and-cut scheme in Algorithm 1.

Finally, the model presented in Section 3.5 can be fully enumerated for most small and medium size instances. However, it is easy to observe that some variables are never used in

Algorithm 1 Pseudocode of the proposed branch-and-cut algorithm

```
At the root node of the search tree, generate and insert all valid inequalities into the program.
Subproblem solution. Solve the LP relaxation of the node.
Termination check:
if there are no more nodes to evaluate then
    Stop.
else
    Select one node from the branch-and-cut tree.
end if
while the solution of the current LP relaxation contains subtours do
    Identify connected components as in Padberg and Rinaldi [29].
    Determine whether the component containing the supplier is weakly connected as in Gendreau
    et al. [13].
    Add violated subtour elimination constraints.
    Subproblem solution. Solve the LP relaxation of the node.
end while
if the solution of the current LP relaxation is integer then
    Go to the termination check.
else
    Branching: branch on one of the fractional variables.
    Go to the termination check.
end if
```

the model, e.g., the ones for which some values of q cannot be obtained by any combination of demands. These variables can be generated and fed to the solver, which will set them to zero in any feasible solution. If one can identify these variables beforehand, it is possible to set them to zero and remove then from the model at a preprocessing phase. Thus, one can (substantially) decrease the size of the model and the memory usage by preprocessing the model and the instance a priori, identifying the subset of variables that should not be generated. We have then implemented a subset sum algorithm to identify all possible values of q from 1 to Q^k that can be achieved by any combination of demands q_i. The ones that are found not to be feasible are not generated and we could reduce the size of the model substantially. Details regarding the improvements provided by this algorithm are presented in Section 5.3.

5 Computational experiments

In this section we provide details on the implementation, benchmark instances, and describe the computational experiments we have performed. Implementation and hardware information is provided in Section 5.1. The description of the existing and new benchmark instances we have used are presented in Section 5.2, followed by the results of our extensive computational experiments in Section 5.3.

5.1 Implementation details

All the formulations described in Section 3 were implemented in C++ and solved with IBM CPLEX Concert Technology 12.5.1. The separation of the subtour elimination constraints was performed with the Concorde package of Applegate et al. [2] and the CVRPSEP package of Lysgaard et al. [25].

We have run all instances described in the next section using all models described in Section 3 with a time limit of three hours and a maximum of 96 Gb of memory. The machines used are all equipped with Intel Xeon™ processors running at 2.66 GHz with 96 GB of RAM installed per node, with the Scientific Linux 6.1 operating system.

5.2 Description of the instances

In order to compare the performance of our models and algorithms, we have used a set of 14 test instances proposed by Salhi and Sari [38] for the MDFSMVRP. These instances were inspired from older benchmarks for other distribution problems proposed by Gillett and Johnson [14], Perl and Daskin [30] and Chao et al. [6]. They are commonly used in the VRP literature. They have been used in previous researches to evaluate the performance of heuristic algorithms, namely the multi-level composite heuristic of Salhi and Sari [38], the variable neighborhood search of Salhi et al. [39], and the hybrid genetic search with advanced diversity control of Vidal et al. [43]. The only lower bounds and solutions obtained with an exact approach existing for these instances were obtained by a branch-bound algorithm applied to a mathematical model presented in Salhi et al. [39].

These instances contain between 50 and 360 customers, and between two and nine depots. There are five vehicle types, i.e., $K = 5$, in all instances. The vehicle capacities are generated centered around the value of the vehicle capacity ($\hat{Q}$) of the original instances designed for of the MDVRP data sets. The vehicle capacities Q^k along with the vehicle variable cost F^k and the vehicle fixed cost α^k are derived based on the following formulas: $Q^k = (0.4 + 0.2k)\hat{Q}$, $F^k = 70 + 10k$ and $\alpha^k = 0.7 + 0.1k$, with $k = 1, \ldots, 5$.

We have also generated ten smaller instances to better evaluate the performance of the different formulations in terms of lower and upper bounds, and of running times. These instances were created by randomly selecting subsets of customers from the smaller instances of Salhi and Sari [38], namely instances 4-55-100 and 4-50-80. Our instances contain two and three depots, from 10 to 30 customers, five vehicle types and different demands distribution. Table 1 contains a list of all instances used in this paper and provides additional information on their origins and sizes.

Table 1: Configurations of the existing and newly generated instances

Instance	Reference	Origin	# depots	# customers	$\hat{Q}$
4-55-100	Salhi and Sari [38]	Perl and Daskin [30]	4	55	100
3-85-100	Salhi and Sari [38]	Perl and Daskin [30]	3	85	100
3-85-160	Salhi and Sari [38]	Perl and Daskin [30]	3	85	160
4-50-80	Salhi and Sari [38]	Gillett and Johnson [14]	4	50	80
4-50-160	Salhi and Sari [38]	Gillett and Johnson [14]	4	50	160
5-75-140	Salhi and Sari [38]	Gillett and Johnson [14]	5	75	140
2-100-100	Salhi and Sari [38]	Gillett and Johnson [14]	2	100	100
2-100-200	Salhi and Sari [38]	Gillett and Johnson [14]	2	100	200
3-100-100	Salhi and Sari [38]	Gillett and Johnson [14]	3	100	100
4-100-100	Salhi and Sari [38]	Gillett and Johnson [14]	4	100	100
2-80-60	Salhi and Sari [38]	Chao et al. [6]	2	80	60
4-160-60	Salhi and Sari [38]	Chao et al. [6]	4	160	60
6-240-60	Salhi and Sari [38]	Chao et al. [6]	6	240	60
9-360-60	Salhi and Sari [38]	Chao et al. [6]	9	360	60
2-10-60	New	Salhi and Sari [38]	2	10	60
2-15-60	New	Salhi and Sari [38]	2	15	60
3-20-80	New	Salhi and Sari [38]	3	20	80
3-25-80	New	Salhi and Sari [38]	3	25	80
3-30-80	New	Salhi and Sari [38]	3	30	80
2-10-60	New	Salhi and Sari [38]	2	10	60
2-15-60	New	Salhi and Sari [38]	2	15	60
3-20-100	New	Salhi and Sari [38]	3	20	100
3-25-100	New	Salhi and Sari [38]	3	25	100
3-30-100	New	Salhi and Sari [38]	3	30	100

5.3 Computational experiments

In this section we describe the results of computational experiments carried out in order to assess the performance of the proposed models and algorithms. Table 2 recalls the configurations of the five formulations tested.

As stated in Section 4, Formulation F5 can be defined only for the values of q that can be attained, which can significantly reduce its size. For the existing instances described in Table 1, the average number of variables of F5 is reduced from 22,395,311 to 19,636,711

Table 2: Summary of the five formulations

Formulation	Objective function and constraints
F1	(1) subject to (2) –(25)
F2	(26) subject to (2)–(6), (7), (9)–(12), (17)–(23), (27)
F3	(28) subject to (2), (4), (29)–(53)
F4	(28) subject to (2), (4), (29)–(34), (38), (39), (54)–(61)
F5	(62) subject to (63)–(79)

when applying the preprocessing step with the subset sum algorithm. We note that for the largest instance of the testbed, 9-360-60, which contains nine depots and 360 customers, the number of required variables could not be enumerated due to memory usage (it required more than 100 Gb of RAM memory). We also observe that the efficiency of the preprocessing phase is highly dependent on the scale and the distribution of the demands. For example, if all demands are multiples of 20, then the number of variables is reduced by almost 20-fold; however, if they are all small and some are unitary, then almost all values of Q^k can be obtained by combining the demands of some customers. In this testbed, the number of generated variables is reduced by more than 16 times for instance 4-55-100, while it remains unchanged for 2-100-100. These values can be observed for all instances in Table 3.

5.3.1 Linear programming relaxation

Solving the linear programming relaxation (LR) can be quite useful as it provides a bound on the optimal value of the integer programs, and it highlights the difference between the formulations. The first experiment we conduct in this section consists of solving the LRs of the five formulations for both data sets with a time limit of 2 hours. We include all the valid inequalities presented in the previous sections. Table 4 summarizes the results of this test. For each model and each solved instance, we provide the LR value and the running time in seconds. In all tables, if an instance cannot be solved we mention NF indicating

Table 3: Number of generated variables for model F5 before and after the preprocessing phase

Instance	Before preprocessing	After preprocessing
4-55-100	7031620	417720
3-85-100	11732160	696960
3-85-160	18701760	998976
4-50-80	4723920	4548960
4-50-160	9389520	9214560
5-75-140	22560000	22400000
2-100-100	10508040	10508040
2-100-200	20912040	20912040
3-100-100	16072635	16072635
4-100-100	21848320	21848320
2-80-60	4101640	4101640
4-160-60	32813120	32813120
6-240-60	110744280	110744280
Average	22,395,311.92	19,636,711.62

not found status, and NC if the number of required variables and constraints could not be enumerated. The results indicate that the LR of model F1 is quite poor. This is due to the drawbacks of this formulation mentioned before, particularly the fact of enumerating the available vehicles. Formulation F1 does not provide a linear relaxation within two hours for any instance of the first data set. Furthermore, comparing the last four formulations substantiates that models F3 and F4 perform extremely well on both data sets compared to model F2. The average of the LR values, only over solved instances (from instance 4-55-100 to 2-80-60), equals 790.94, 1554.14, and 1649.38 while the average running time is increasing from 55 to 77 and to 574 seconds for models F2, F3 and F4, respectively. The average computation time of model F4 is almost seven times the average computation time of model F3 whereas the difference between the LRs of these two formulations is small. This implies that disaggregating loading variables requires more computational time to find slightly better relaxations. Model F5 provides better LR values for all four solved instances compared to all other formulations. Over all models, the average time taken to solve the LRs is not negligible. This is due to the high number of variables and constraints required to model the problem.

Table 4: Linear programming relaxations for the five formulations

Instance	Formulation F1		Formulation F2		Formulation F3		Formulation F4		Formulation F5	
	Value	Time(s)	Value	Time(s)	Value	Time(s)	Value	Time(s)	Value	Time(s)
4-55-100	NF	7200	574.32	17	1313.54	29	1354.41	58	**1359.99**	56
3-85-100	NF	7200	841.57	43	2027.98	66	2079.01	456	**2094.42**	157
3-85-160	NF	7200	631.65	44	1347.39	77	1411.31	505	**1435.19**	357
4-50-80	NF	7200	642.90	23	1322.30	21	1381.63	58	**1416.05**	4348
4-50-160	NF	7200	496.07	10	807.47	17	**890.99**	89	NF	7200
5-75-140	NC	7200	707.52	62	1362.47	102	**1478.41**	530	NF	7200
2-100-100	NC	7200	966.10	53	2095.84	94	**2191.73**	816	NF	7200
2-100-200	NC	7200	749.49	54	1262.37	80	**1382.34**	1021	NF	7200
3-100-100	NC	7200	952.52	84	1990.05	143	**2106.65**	1074	NF	7200
4-100-100	NC	7200	951.76	166	1993.18	197	**2103.29**	1422	NF	7200
2-80-60	NC	7200	1186.35	48	1573.03	24	1763.45	287	**1790.17**	6861
4-160-60	NC	7200	NF	7200	**3063.71**	1090	NF	7200	NF	7200
6-240-60	NC	7200	NF	7200	NF	7200	NF	7200	NF	7200
9-360-60	NC	7200	NF	7200	NF	7200	NF	7200	NC	7200
2-10-60	268.51	5	268.58	0	399.87	0	409.55	0	**422.33**	3
2-15-60	355.68	50	356.02	0	600.23	0	627.41	1	**650.23**	9
3-20-80	333.43	540	334.30	0	594.47	1	626.40	2	**640.66**	117
3-25-80	395.63	1880	397.93	1	706.59	1	744.38	6	**759.54**	218
3-30-80	NF	7200	438.25	2	828.87	3	866.25	7	**884.23**	393
2-10-60	245.29	3	246.38	0	448.59	0	459.77	0	**468.59**	0
2-15-60	316.37	43	318.53	0	646.36	0	659.91	1	**681.01**	0
3-20-100	255.55	206	255.41	1	525.82	1	544.05	2	**544.71**	2
3-25-100	303.23	1299	303.47	1	631.16	1	658.49	4	**660.80**	3
3-30-100	382.75	2216	383.20	1	784.34	2	820.20	9	**824.98**	5

5.3.2 Comparison of upper and lower bounds

We now present the computational results of the solutions we have obtained when applying branch-and-bound and branch-and-cut for the five proposed formulations. Table 5 summarizes the results after three hours of running time with CPLEX. We report the upper bound (UB) and the lower bound (LB) of each formulation for each instance, if they are found. We provide the average percentage gaps over the two testbeds. The percentage gap is given by the ratio ($\frac{UB-LB}{UB}100$). We also give the average time in seconds spent to solve the new testbed. Bold face is used to indicate the best results.

A deeper analysis of the formulations highlights a remarkable improvement over all the lower bounds and the number of solved instances compared to the LRs results. The results clearly show that formulation F1 is outperformed by all the other formulations, even on small instances. The largest instance size that can be solved by formulation F1 is 4-5-160. Model F1 could identify a feasible solution only for three (out of 14) instances, whereas formulation F2 is able to solve all the instances of the two testbeds. This implies that the compact formulation, reducing the number of generated variables, has a positive impact on the model performance. Model F2 provides tighter bounds compared to F1 but is still uncompetitive compared to the other formulations. The results of Table 5 distinctly show the performance of the last three formulations to solve the MDFSMVRP. Model F4 could generally provide better bounds compared to all other formulations, especially on the first testbed, despite the fact that model F3 yields better UBs. We observe that there is a difference between models F3 and F4 regarding the overall gaps. The solutions provided by F4 are 8.2% and 1.2% better than the solutions provided by F3 on the two testbeds, respectively. Model F4 provides eight best LBs and five best UBs over 14 instances, while F3 provides eight best UBs on the first testbed. This implies that disaggregating the commodity flow variables is likely improving the model performance. Model F5 has better bounds on the first three instances compared to all other formulations and provides the best gap for instances 2-100-100 and 2-80-60. This is due to the fact that few variables are generated in these test instances, characterized by a regular distribution of customers demands and/or a small number of customers. However, even if model F5 provides six best LBs over 14 instances, its overall average gap is about three times the overall average gap of model F4. Regarding the small generated instances, F5 outperforms all the other formulations and provides eight optimal solutions over 10, with an average gap equal to 0.57% and an average running time of 2390 seconds. F4 provides competitive solutions with slightly better average gap (0.49%) than F5 within less computation time (1832 seconds). However, formulation F4 proves the optimality only for the smallest instance with two depots and 10 customers. The computation times and the average gaps provided by formulations F1 and F2 on these small instances are quite high. In particular they require, on average, 8590 and 8841 seconds to solve instances with up to three depots and 30 customers. These results point out again that formulations F3, F4 and F5 are the most suitable among the five proposed to solve small, medium and large size instances of the MDFSMVRP.

A transversal analysis over Tables 4 and 5 allows us to remark that the deductions derived after solving the models with integrality restrictions confirm the preliminary results derived from the LR experiment. In addition, we observe that, on average, the values of the LR of models F3 and F4 over the first 11 solved instances in Table 4 is equal to almost 0.7 and 0.8 times the UBs of the 3-hour execution of these models. Thereby, one can conclude that the proposed models, especially the commodity flow formulations are good enough as they provide strong linear relaxations. Finally, we can derive some comments on the relative difficulty of the problem. We observe that the average gaps remain large, especially on instances with more than two depots and 100 customers.

5.3.3 Comparison against the best known solutions

As it was mentioned, the literature devoted to the MDFSMVRP is rather scarce and the published works on this specific variant are focused on heuristic methods. We are only aware of the exact bounds recently obtained by the three-hour CPLEX execution of Salhi et al. [39]. The performance of the proposed formulations is assessed with respect to the available lower bounds provided in Salhi et al. [39] and to the best upper bounds given heuristically by Vidal et al. [43].Table 6 presents the results of the best formulations proposed in Section 3 compared to the state-of-the art methods. For completeness, we have also reported the percentage gap between the best LBs and UBs obtained over the proposed formulations in the column *Best gap (%).* The results in Table 6 show that the proposed formulations could often identify a feasible solution for all the instances, even for the largest instance considered with nine depots and 360 customers, unlike the exact solution method of Salhi et al. [39]. The largest instance solved by this method includes four depots and 100 customers. Models F3 and F4 yield better optimality gaps than that work on all instances. Note also that the improvement with respect to the bounds given by Salhi et al. [39] are significant. We have improved all the LBs and UBs with respect

Table 5: Summary of computational results obtained from the five formulations

Instance	Formulation F1			Formulation F2			Formulation F3			Formulation F4			Formulation F5		
	UB	LB	Gap (%)	UB	LB	Gap (%)	UB	LB	Gap (%)	UB	LB	Gap (%)	UB	LB	Gap (%)
4-55-100	1612.30	552.52	65.73	1606.63	575.62	64.17	1438.50	1328.99	7.61	1408.25	1361.85	3.29	**1398.30**	**1384.55**	0.98
3-85-100	NF	NF	-	2258.22	842.50	62.69	2209.95	2040.45	7.67	2163.63	2086.32	3.57	**2131.46**	**2116.12**	0.72
3-85-160	NF	NF	-	1698.58	633.22	62.72	1675.05	1360.58	18.77	1663.01	1423.30	14.41	**1483.51**	**1451.86**	2.13
4-50-80	1770.66	602.44	65.98	1770.66	621.48	64.90	1573.68	1348.51	14.31	**1565.27**	1390.51	11.16	1733.20	**1416.09**	18.30
4-50-160	1217.74	485.68	60.12	1216.52	499.13	58.97	**1021.59**	832.21	18.54	1122.08	**907.71**	19.10	1217.74	902.18	25.91
5-75-140	NC	NC	-	1962.67	708.06	63.92	1831.02	1388.41	24.15	**1828.73**	**1483.11**	18.92	NF	NF	-
2-100-100	NC	NC	-	2757.95	968.67	64.88	**2660.50**	2116.36	20.45	2725.87	2201.16	19.25	2757.95	**2236.91**	18.89
2-100-200	NC	NC	-	1826.28	750.27	58.92	**1818.69**	1280.61	29.59	1825.91	**1396.05**	23.54	NF	NF	-
3-100-100	NC	NC	-	2728.89	953.12	65.07	**2648.17**	2013.95	23.95	2696.05	**2109.31**	21.76	NF	NF	-
4-100-100	NC	NC	-	2691.82	952.05	64.63	**2626.33**	2003.95	23.70	2684.89	**2104.36**	21.63	NF	NF	-
2-80-60	NC	NC	-	2573.82	1186.35	53.91	2566.13	1683.51	34.39	**2565.53**	1788.35	30.29	2571.35	**1794.38**	30.22
4-160-60	NC	NC	-	5172.65	2199.05	57.49	**5157.49**	3094.10	40.01	5172.65	**3506.89**	32.20	NF	NF	-
6-240-60	NC	NC	-	7758.97	3182.68	58.98	**7758.97**	4564.18	41.18	**7758.97**	**5243.12**	32.43	NF	NF	-
9-360-60	NC	NC	-	11638.50	4441.25	61.84	**11638.50**	0.00	100.00	**11638.50**	**7852.44**	32.53	NC	NC	-
Average gap			90.99			61.65			28.88			20.63			56.94
2-10-60	**441.59**	**441.59**	0.00	**441.59**	441.57	0.00	**441.59**	**441.59**	0.00	**441.59**	**441.59**	0.00	**441.59**	**441.59**	0.00
2-15-60	679.18	611.92	9.90	679.18	489.20	27.97	**674.32**	674.26	0.01	**674.32**	674.26	0.01	**674.32**	**674.32**	0.00
3-20-80	784.97	354.03	54.90	757.34	405.73	46.43	**666.02**	665.95	0.01	**666.02**	665.97	0.01	**666.02**	**666.02**	0.00
3-25-80	845.41	409.43	51.57	839.13	429.99	48.76	794.46	745.08	6.21	**787.37**	787.29	0.01	**787.37**	**787.30**	0.01
3-30-80	1009.93	448.00	55.64	1005.05	448.83	55.34	932.13	870.77	6.58	932.13	**890.71**	4.44	**940.49**	886.95	5.69
2-10-60	**482.09**	**482.09**	0.00	**482.09**	482.05	0.01	**482.09**	**482.09**	0.00	**482.09**	**482.09**	0.00	**482.09**	**482.09**	0.00
2-15-60	**690.38**	**690.38**	0.00	**690.38**	393.98	42.93	**690.38**	690.31	0.01	**690.38**	690.31	0.01	**690.38**	**690.38**	0.00
3-20-100	597.44	543.12	9.09	605.95	293.85	51.51	**563.19**	563.13	0.01	**563.19**	563.15	0.01	**563.19**	**563.19**	0.00
3-25-100	727.54	651.16	10.50	717.86	321.37	55.23	**676.59**	674.68	0.28	**676.59**	676.53	0.01	**676.59**	**676.59**	0.00
3-30-100	994.17	735.29	26.04	994.17	397.68	60.00	**839.98**	812.71	3.25	**839.98**	836.77	0.38	**839.98**	**839.98**	0.00
Average gap/ time(s)			21.76/8590.10			38.82/8811.400			1.64/5138.40			0.49/1832.56			0.57/2390.00

to that work. The average LB is increased by 8.63% for the first eleven instances solved by Salhi et al. [39], and the average UB is reduced by 21.30%. One particular example is that of instance 3-85-160 for which the gap was 47.29% and is now just 2.13%. The average gap over all the instances of the first testbed has decreased from 51.00% to 17.84%. The comparison of our best results against the heuristic of Vidal et al. [43] show that we could not improve the UBs found heuristically but our gaps are tight. Even though the quality of the UBs is not improved, the introduction of these different formulations helps providing very good lower bounds.

5.3.4 Effect of valid inequalities

We now briefly analyze the effect of the valid inequalities proposed for each model. We study the effect of the whole set of valid inequalities in each model on each upper and lower bound. We have decided not to study the impact of each valid constraint proposed in each model because this would lead to a combinatorial and unmanageable comparison between valid inequalities, which is not the focus of this paper. We have compared the average gaps between two configurations of each formulation, without and with valid inequalities, on each testbed, with the maximum computing time limit of three hours. Table 7 summarizes these results. On average, they clearly show the benefits of using valid inequalities especially for the explicit formulation. The average gap is reduced by almost 50% on the new smaller instances. We can also observe that the introduction of valid inequalities is more relevant for formulations which explicit the index of the vehicle because it is hard to generate efficient valid inequalities for variables that do not carry at least the vehicle type traversing an arc, as it is the case of formulation F3.

Table 6: Comparison of the best formulations with the state-of-the art methods

Instance	Vidal et al. [43]	Salhi et al. [39]			Formulation F3			Formulation F4			Formulation F5			F3, F4, F5
	UB	UB	LB	Gap (%)	UB	LB	Gap (%)	UB	LB	Gap (%)	UB	LB	Gap (%)	Best gap (%)
4-55-100	-	1621.90	1299.80	19.86	1438.50	1328.99	7.61	1408.25	1361.85	3.29	**1398.30**	**1384.55**	0.98	0.98
3-85-100	-	2677.80	1996.00	25.46	2209.95	2040.45	7.67	2163.63	2086.32	3.57	**2131.46**	**2116.12**	0.72	0.72
3-85-160	-	2516.00	1326.10	47.29	1675.05	1360.58	18.77	1663.01	1423.30	14.41	**1483.51**	**1451.86**	2.13	2.13
4-50-80	**1477.73**	1725.20	1318.50	23.57	1573.68	1348.51	14.31	1565.27	1390.51	11.16	1733.20	**1416.09**	18.30	9.53
4-50-160	**957.73**	1378.90	799.70	42.00	1021.59	832.21	18.54	1122.08	**907.71**	19.10	1217.74	902.18	25.91	11.14
5-75-140	**1569.67**	2561.10	1341.40	47.62	1831.02	1388.41	24.15	1828.73	**1483.11**	18.92	NF	NF	-	18.90
2-100-100	**2292.64**	3039.70	2079.30	31.59	2660.50	2116.36	20.45	2725.87	2201.16	19.25	2757.95	**2236.91**	18.89	15.92
2-100-200	**1453.64**	2265.60	1228.30	45.78	1818.69	1280.61	29.59	1825.91	**1396.05**	23.54	NF	NF	-	23.23
3-100-100	**2208.66**	3390.30	1970.20	41.89	2648.17	2013.95	23.95	2696.05	**2109.31**	21.76	NF	NF	-	20.34
4-100-100	**2198.91**	3609.90	1971.30	45.39	2626.33	2003.95	23.70	2684.89	**2104.36**	21.63	NF	NF	-	19.87
2-80-60	**2072.18**	2846.40	1607.60	43.52	2566.13	1683.51	34.39	2565.53	1788.35	30.29	2571.35	**1794.38**	30.22	30.05
4-160-60	**3973.47**	NF	3032.60	-	5157.49	3094.10	40.01	5172.65	**3506.89**	32.20	NF	NF	-	32.00
6-240-60	**5887.43**	NF	NF	-	7758.97	4564.18	41.18	7758.97	**5243.12**	32.43	NF	NF	-	32.42
9-360-60	**8709.26**	NF	NF	-	11638.50	0.00	100.00	11638.50	**7852.44**	32.53	NC	NC	-	32.53
Average gap				51.00			28.88			20.63			56.94	17.84

Table 7: Average percentage gaps with and without valid inequalities

Instances	Formulation F1		Formulation F2		Formulation F3		Formulation F4		Formulation F5	
	Without VI	With VI	Without VI	With VI	Without VI	With VI	Without VI	With VI	Without VI	With VI
Existing	94.66	90.99	61.79	61.65	29.45	28.88	25.52	20.29	62.43	56.94
New	57.68	21.76	40.76	38.82	2.05	1.64	0.69	0.49	1.18	0.57

6 Conclusions

In this paper we have modeled and solved the MDFSMVRP. We have presented five different formulations for this difficult distribution problem. The first one is a three-index VRP formulation with an explicit vehicle index, and the second one is more compact, in which individual vehicles are not explicitly identified. The third and the fourth models are commodity flow formulations without a vehicle index. They are based on loading variables to model capacity and connectivity requirements. The fifth and last model is a capacity-indexed formulation, which is a based more compact single commodity flow. We have proposed several valid inequalities to strengthen the formulations and we have solved them by branch-and-cut and by branch-and-bound.

We compared the bounds of these formulations on existing instances and on newly generated ones. The results show that the commodity flow formulations and the capacity-indexed formulation provide better bounds. Our results also show that compact formulations represent a very promising research avenue. On classical benchmark instances our methods could improve all lower bounds, and we have obtained the best upper bounds and gaps when compared to another exact method from the literature.

References

[1] Y. Adulyasak, J.-F. Cordeau, and R. Jans. Formulations and branch-and-cut algorithms for multivehicle production and inventory routing problems. *INFORMS Journal on Computing*, 26(1):103–120, 2013.

[2] D. L. Applegate, R. E. Bixby, V. Chvátal, and W. J. Cook. *The Traveling Salesman Problem: A Computational Study*. Princeton University Press, 2011.

[3] R. Baldacci, M. Battarra, and D. Vigo. Routing a heterogeneous fleet of vehicles. In B. L. Golden, S. Raghavan, and E. A. Wasil, editors, *The Vehicle Routing Problem: Latest Advances and New Challenges*, pages 3–27. Springer, New York, 2008.

[4] A. Bettinelli, A. Ceselli, and G. Righini. A branch-and-cut-and-price algorithm for the multi-depot heterogeneous vehicle routing problem with time windows. *Transportation Research Part C: Emerging Technologies*, 19(5):723–740, 2011.

[5] R. Bosch and M. Trick. Integer programming. In E. K. Burke and G. Kendall, editors, *Search Methodologies*, pages 69–95. Springer, London, 2005.

[6] I. M. Chao, B. L. Golden, and E. A. Wasil. A new heuristic for the multi-depot vehicle routing problem that improves upon best-known solutions. *American Journal of Mathematical and Management Sciences*, 13(3-4):371–406, 1993.

[7] L. C. Coelho and G. Laporte. The exact solution of several classes of inventory-routing problems. *Computers & Operations Research*, 40(2):558–565, 2013.

[8] L. C. Coelho and G. Laporte. Improved solutions for inventory-routing problems through valid inequalities and input ordering. *International Journal of Production Economics*, 155:391–397, 2014.

[9] L. C. Coelho, J. Renaud, and G. Laporte. Road-based goods transportation: A survey of real-world applications from 2000 to 2015. Technical Report FSA-2015-007, Québec, Canada, 2015.

[10] G. Dantzig, R. Fulkerson, and S. Johnson. Solution of a large-scale traveling-salesman problem. *Journal of the Operations Research Society of America*, 2:393–410, 1954.

[11] W. M. Garvin, H. W. Crandall, J. B. John, and R. A. Spellman. Applications of linear programming in the oil industry. *Management Science*, 3(4):407–430, 1957.

[12] B. Gavish and S. Graves. Scheduling and routing in transportation and distribution systems: formulations and new relaxations. *Rochester: Graduate School of Management, University of Rochester*, 1982.

[13] M. Gendreau, G. Laporte, and F. Semet. The covering tour problem. *Operations Research*, 45(4):568–576, 1997.

[14] B. E. Gillett and J. G. Johnson. Multi-terminal vehicle-dispatch algorithm. *Omega*, 4(6):711–718, 1976.

[15] M. T. Godinho, L. Gouveia, and T. L. Magnanti. Combined route capacity and route length models for unit demand vehicle routing problems. *Discrete Optimization*, 5 (2):350–372, 2008.

[16] S. Irnich, M. Schneider, and D. Vigo. Four variants of the vehicle routing problem. In P. Toth and D. Vigo, editors, *Vehicle Routing: Problems, Methods, and Applications*, pages 241–260. MOS-SIAM Series on Optimization, Philadelphia, 2014.

[17] I. Karaoglan, F. Altiparmak, I. Kara, and B. Dengiz. The location-routing problem with simultaneous pickup and delivery: Formulations and a heuristic approach. *Omega*, 40(4):465–477, 2012.

[18] Ç. Koç, T. Bektaş, O. Jabali, and G. Laporte. The fleet size and mix location-routing problem with time windows: Formulations and a heuristic algorithm. Technical Report CIRRELT-2015-04, Montreal, Canada, 2015.

[19] Ç. Koç, T. Bektaş, O. Jabali, and G. Laporte. Thirty years of heterogeneous vehicle routing. *European Journal of Operational Research*, 2015, forthcoming.

[20] R. Lahyani, L. C. Coelho, M. Khemakhem, G. Laporte, and F. Semet. A multi-compartment vehicle routing problem arising in the collection of olive oil in Tunisia. *Omega*, 51:1–10, 2015.

[21] R. Lahyani, M. Khemakhem, and F. Semet. Rich vehicle routing problems: From a taxonomy to a definition. *European Journal of Operational Research*, 241(1):1–14, 2015.

[22] G. Laporte. Generalized subtour elimination constraints and connectivity constraints. *Journal of the Operational Research Society*, 37(5):509–514, 1986.

[23] G. Laporte. Fifty years of vehicle routing. *Transportation Science*, 43(4):408–416, 2009.

[24] G. Laporte and Y. Nobert. Exact algorithms for the vehicle routing problem. In S. Martello, G. Laporte, M. Minoux, and C. Ribeiro, editors, *Surveys in Combinatorial Optimization*, pages 147–184. North-Holland Mathematics Studies, North-Holland, 1987.

[25] J. Lysgaard, A. N. Letchford, and R. W. Eglese. A new branch-and-cut algorithm for the capacitated vehicle routing problem. *Mathematical Programming*, 100(2): 423–445, 2004.

[26] S. Mancini. A real-life multi depot multi period vehicle routing problem with a heterogeneous fleet: Formulation and adaptive large neighborhood search based matheuristic. *Transportation Research Part C: Emerging Technologies*, 2015, forthcoming.

[27] J. R. Montoya-Torres, J. L. Franco, S. N. Isaza, H. F. Jiménez, and N. Herazo-Padilla. A literature review on the vehicle routing problem with multiple depots. *Computers & Industrial Engineering*, 79:115–129, 2015.

[28] D. Naddef and G. Rinaldi. Branch-and-cut algorithms for the capacitated vehicle routing problem. In P. Toth and D. Vigo, editors, *The Vehicle Routing Problem*, pages 53–84. SIAM Monographs on Discrete Mathematics and Applications, Philadelphia, 2002.

[29] M. W. Padberg and G. Rinaldi. A branch-and-cut algorithm for the resolution of large-scale symmetric traveling salesman problems. *SIAM Review*, 33(1):60–100, 1991.

[30] J. Perl and M. S. Daskin. A warehouse location-routing problem. *Transportation Research Part B: Methodological*, 19(5):381–396, 1985.

[31] A. Pessoa, M. V. S. Poggi de Aragão, and E. Uchoa. A robust branch-cut-and-price algorithm for the heterogeneous fleet vehicle routing problem. *Lecture Notes in Computer Science*, 4525:150–160, 2007.

[32] A. Pessoa, M. V. S. Poggi de Aragão, and E. Uchoa. Robust branch-cut-and-price algorithms for vehicle routing problems. In B. L. Golden, S. Raghavan, and E. A. Wasil, editors, *The Vehicle Routing Problem: Latest Advances and New Challenges*, pages 297–325. Springer, New York, 2008.

[33] A. Pessoa, E. Uchoa, and M. V. S. Poggi de Aragão. A robust branch-cut-and-price algorithm for the heterogeneous fleet vehicle routing problem. *Networks*, 54(4): 167–177, 2009.

[34] J. C. Picard and M. Queyranne. The time-dependent traveling salesman problem and its application to the tardiness problem in one-machine scheduling. *Operations Research*, 26(1):86–110, 1978.

[35] M. V. S. Poggi de Aragão and E. Uchoa. New exact algorithms for the capacitated cehicle routing problem. In P. Toth and D. Vigo, editors, *Vehicle Routing: Problems, Methods, and Applications*, pages 59–86. MOS-SIAM Series on Optimization, Philadelphia, 2014.

[36] A. Rahimi-Vahed, T. G. Crainic, M. Gendreau, and W. Rei. Fleet-sizing for multi-depot and periodic vehicle routing problems using a modular heuristic algorithm. *Computers & Operations Research*, 53:9–23, 2015.

[37] S. Salhi and G. K. Rand. Incorporating vehicle routing into the vehicle fleet composition problem. *European Journal of Operational Research*, 66(3):313–330, 1993.

[38] S. Salhi and M. Sari. A multi-level composite heuristic for the multi-depot vehicle fleet mix problem. *European Journal of Operational Research*, 103(1):95–112, 1997.

[39] S. Salhi, A. Imran, and N. A. Wassan. The multi-depot vehicle routing problem

with heterogeneous vehicle fleet: Formulation and a variable neighborhood search implementation. *Computers & Operations Research*, 52:315–325, 2014.

[40] H. D. Sherali and J. C. Smith. Improving discrete model representations via symmetry considerations. *Management Science*, 47(10):1396–1407, 2001.

[41] P. Toth and D. Vigo. An overview of vehicle routing problems. In P. Toth and D. Vigo, editors, *The Vehicle Routing Problem*, pages 1–26. SIAM Monographs on Discrete Mathematics and Applications, Philadelphia, 2002.

[42] P. Toth and D. Vigo. *Vehicle Routing: Problems, Methods, and Applications*. MOS-SIAM Series on Optimization, Philadelphia, 2014.

[43] T. Vidal, T. G. Crainic, M. Gendreau, and C. Prins. Implicit depot assignments and rotations in vehicle routing heuristics. *European Journal of Operational Research*, 237(1):15–28, 2014.

[44] Y. Xu and W. Jiang. An improved variable neighborhood search algorithm for multi depot heterogeneous vehicle routing problem based on hybrid operators. *International Journal of Control and Automation*, 7(3):299–316, 2014.

[45] H. Yaman. Formulations and valid inequalities for the heterogeneous vehicle routing problem. *Mathematical Programming*, 106(2):365–390, 2006.