INTERNATIONAL JOURNAL OF LOGISTICS: RESEARCH AND APPLICATIONS
https://doi.org/10.1080/13675567.2019.1630374

Multi-depot vehicle routing problem: a comparative study of alternative formulations

Tânia Rodrigues Pereira Ramos [a,b], Maria Isabel Gomes [c] and Ana Paula Barbosa Póvoa [a]

[a]Centre of Management Studies – Instituto Superior Técnico (CEG-IST), Universidade de Lisboa, Lisbon, Portugal; [b]Business Research Unit – ISCTE (BRU-ISCTE), Instituto Universitário de Lisboa, Lisbon, Portugal; [c]Centro de Matemática e Aplicações – Faculdade de Ciências e Tecnologia (CMA-FCT), Universidade Nova de Lisboa, Lisbon, Portugal

ABSTRACT

This paper proposes a two-commodity flow formulation for the MDVRP considering a heterogeneous vehicle fleet and maximum routing time. Its computational performance is compared to a traditional formulation, the so called three-index formulation, which is adapted to fit the same problem. With the creation of subtours being a problem within such formulations, four alternative sets of constraints are considered to eliminate them: the Dantzig–Fulkerson–Johnson constraint, the Miller–Tucker–Zemlin constraint, the transit load constraint, and the arrival time constraint. The resulting mixed-integer linear programming models are then applied to a number of benchmark instances and the obtained performance results are compared.

ARTICLE HISTORY
Received 10 November 2017
Accepted 30 May 2019

KEYWORDS
Multi-depot; vehicle routing; MILP; subtour elimination

1. Introduction

The multi-depot vehicle routing problem (MDVRP) arises as a generalisation of the vehicle routing problem (VRP), where vehicles depart from and return to one of multiple depot locations. Therefore, besides the definition of the vehicles' routes, it is also necessary to decide from which depot the customers are visited. The MDVRP simultaneously defines the service areas of each depot and establishes the associated vehicle routes. The MDVRP is one of several variants of the routing problems (see, for example, a taxonomic review for routing problems in the work of Braekers, Ramaekers, and Nieuwenh 2016). Different mathematical formulations for the VRP have been proposed in the literature, namely: (1) the two-index vehicle flow formulation (Laporte, Nobert, and Desrochers 1985), where x_{ij} is an integer variable representing the number of times edge (i,j) appears in the optimal solution; (2) the three-index vehicle flow formulation (Golden, Magnanti, and Nguyen 1977), where x_{ijk} is a binary variable representing the routing solution: $x_{ijk}=1$, if edge (i,j) is traversed by vehicle k; $x_{ijk}=0$, otherwise; (3) the two-commodity flow formulation (Baldacci, Hadjiconstantinou, and Mingozzi 2004) where x_{ij} is a binary variable equal to 1 if edge (i,j) is in the solution and y_{ij} and y_{ji} are the flow variables representing, respectively, the load and the empty space on the vehicle when the edge (i,j) is traversed; (4) and the set partitioning formulation (Baldacci, Christofides, and Mingozzi 2008; Balinski and Quandt 1964) where x_j is a binary variable associated with each feasible route j, being equal to 1 if route j is in the optimal solution.

The MDVRP can be defined by a graph $G=(V,A)$, where $V=\{1,\ldots,n+w\}$ is the vertex set and $A=\{(i,j):i,j\in V, i\neq j\}$ the edge set. The vertex V is partitioned into two subsets $V_c=\{1,\ldots,n\}$ and $V_d=\{n+1,\ldots,n+w\}$ representing, respectively, the set of n places to

CONTACT Tânia Rodrigues Pereira Ramos tania.p.ramos@tecnico.ulisboa.pt

visit and the set of w depots. At each depot, a maximum of r vehicles of capacity Q are available. Each vertex $i \in V_c$ has a nonnegative demand p_i and a nonnegative service duration t_i. A distance matrix $D = [d_{ij}]$ is associated to set A. In a MDVRP, one wants to define at minimum cost the set of vehicle routes such that: (1) each route starts and ends at the same depot, (2) each customer is visited exactly once by a single vehicle, (3) route demand cannot exceed the vehicle capacity, (4) route duration (including travel and service times) does not exceed a preset limit.

This paper proposes a new flow formulation for the MDVRP based on a two-commodity flow formulation. In order to conclude on the proposed formulation developed here, it also adapts a three-index formulation and compares them both in terms of computational results. Keeping in mind that one of the existing problems in VRP formulations is the elimination of subtours, we also explore different constraint formulations to avoid subtours, within the three-index formulation, so as to be able to evaluate the quality of the new developed formulation in comparison to other more traditional formulations available in the literature, if a commercial solver is used as a 'black box'. Among other reasons for using a solver as a 'black box' we should point out the difficulty of implementing the majority of algorithms and more complex formulations proposed by related literature when solving real-world management problems.

This paper is structured as follows: After a literature review on multi-depot vehicle routing problem (MDVRP) in Section 2, the proposed two-commodity flow formulation for the MDVRP and the three-index formulation adapted to the MDVRP are presented (Section 3). In Section 4, some small to medium size benchmarking instances are solved, and the computational results of the given formulations are compared. Section 5 presents some conclusions and future work directions.

2. Literature review

The MDVRP appears as a generalisation of the classical Capacitated Vehicle Routing Problem (CVRP). The CVRP is a hard combinatorial problem, and it is usually tackled by means of heuristics (Cordeau, Gendreau, and Laporte 1997). However, with the continuous developments of integer programming software systems, the capability to solve larger problem instances has improved considerably (see Atamturk and Savelsbergh (2005) for a solvers' state-of-the-art review). Baldacci, Toth, and Vigo (2007) reviewed recent exact methods for the CVRP and presented and compared different mathematical formulations of CVRP proposed in the literature: the two and three-index vehicle flow formulation; the two-commodity flow formulation and the set partitioning formulation. Later on, Fukasawa et al. (2006) and Baldacci, Christofides, and Mingozzi (2008) also present exact methods to solve the CVRP. The methods are applied to a set of instances with a number of customers between 12 and 199, making use of up to 14 vehicles.

For the MDVRP, there are still few exact algorithms available in the literature, while several heuristic procedures have already been proposed. According to the literature review on the MDVRP done by Montoya-Torres et al. (2015), 25% of the works on MDVRP employed exact solution techniques, while the remaining 75% of the works employed heuristics or meta-heuristics techniques.

Regarding exact algorithms, Laporte, Nobert, and Arpin (1984, 1988) have developed branch and bound algorithms for solving the symmetric and asymmetric version of the MDVRP. More recently, Baldacci and Mingozzi (2009) have developed an exact method for solving the Heterogeneous Vehicle Routing Problem (HVRP) that is capable of solving, among other problems, the MDVRP. This algorithm is based on the set partitioning formulation, where firstly a procedure is applied to generate routes, followed by three bounding procedures to reduce the number of variables. In the set partitioning formulation, variables represent the previously generated routes. The authors present the computational results for MDVRP instances with up to 200 customers and 2–5 depots. Later on, Bettinelli, Ceselli, and Righini (2011) proposed a branch-and-cut-and-price algorithm for the multi-depot heterogeneous vehicle routing problem with time windows, considering different combinations of cutting and pricing strategies. The results of three different sets of experiments over three datasets of instances were explored. Contardo and Martinelli (2014) presented a new exact method for the MDVRP based on variable fixing, column-and-cut generation and column enumeration. The proposed method was able to prove optimality for the first time for some benchmarking instances.

In terms of heuristic procedures, one of the first works was proposed by Tillman and Cain (1972). The authors presented a savings method with a modified distance metric allowing for multiple depots. Golden, Magnanti, and Nguyen (1977) propose two heuristic algorithms. In the first one, customers are assigned to depots and while simultaneously defining routes. In the second one, customers are first classified as borderline or non-borderline node based on the ratio $r(i)$ proposed by Gillet and Johnson (1976). The non-borderline nodes then are assigned to the nearest depot while for the borderline nodes the first algorithm is applied to decide their assignment. Lastly, routes are defined through a heuristic algorithm developed for the VRP. Chao, Golden, and Wasil (1993) develop a heuristic approach based on Dueck's (1993) record-to-record method to the MDVRP. Renaud, Laporte, and Boctor (1996) developed the FIND algorithm, based on tabu search, to improve the initial solution found by the heuristic Improved Petal after assigning each customer to the nearest depot. A tabu search heuristic is also developed by Cordeau, Gendreau, and Laporte (1997) that is capable to solve the MDVRP, PVRP and PTSP. The authors show the MDVRP can be formulated as a special case of the PVRP and, therefore, the same methodology can be used to solve both problems. Salhi and Sari (1997) propose a multi-level heuristic to solve the multi-depot vehicle fleet mix problem (MDVFM) where, besides the decisions addressed by a multi-depot vehicle routing problem, the composition of the vehicle fleet is also considered. The heuristic is tested on benchmark problems for the MDVRP and good results were achieved. Lim and Wang (2005) propose two solution methodologies for the multi-depot vehicle routing problem with fixed vehicles distribution (MDVRPFD). In the first approach, two rules (urgency and group assignment) assign all customers to depots, followed by a TSP algorithm that solves the sequencing problem for each depot. In the second approach assignment and routing problems are solved simultaneously, since customers are assigned to depots while routes are being established. At each iteration, the customer with the least cost insertion is selected and added to a route, which is then rearranged through a tabu search algorithm. Crevier, Cordeau, and Laporte (2007) propose a solution methodology to solve the multi-depot vehicle routing problem with inter-depots routes, where several techniques are applied: adaptive memory principle, tabu search method and integer linear programming. The problem is decomposed into three subproblems: MDVRP, VRP and Inter-Depot and routes are generated for each subproblem through the tabu search algorithm propose by Cordeau, Gendreau, and Laporte (1997). A set partitioning algorithm is then applied, in order to select the least cost feasible rotations (the authors define rotation as the set of routes assigned to a vehicle). Ho et al. (2008) also propose two hybrid genetic algorithms for the MDVRP with different procedures to find an initial solution. Dondo and Cerda (2009) present a local search improvement algorithm for the MDVRP with Time-Windows, based on mixed integer linear models (MILP). Yucenur and Demirel (2011) propose a geometric shape-based genetic clustering algorithm for the assignment of clients to the depot in the MDVRP. Gulczynski, Golden, and Wasil (2011) propose a new variant of the MDVRP; the multi-depot split delivery vehicle routing problem, and develop an integer programming-based heuristic to solve it. Vidal et al. (2012) proposed a genetic-based metaheuristic for three VRP problems, including the MDVRP. The authors propose new methodologies to define several population operators and are able to find a new best solution for some known problem instances. Yu et al. (2013) addressed the dynamic MDVRP through an ant colony algorithm. Salhi, Imran, and Wassan (2014) presented a flow-based formulation for the MDVRP with an heterogeneous fleet and developed a Variable Neighbourhood Search algorithm capable of obtaining 23 new best solutions out of 26 benchmarking instances. Escobar et al. (2014) develop a hybrid granular tabu search for the MDVRP. Zhang et al. (2015) use a new-generation solver, the LocalSolver, to model and solve the MDVRP with time-windows in which the vehicles' cargo may be replenished along their trips. Chávez, Escobar, and Echeverri (2016) propose an ant colony algorithm for the MDVRP with backhauls and consider three different objective functions: distance travelled, travelling times and total consumption of energy. Li, Li, and Pardalos (2016) developed a hybrid genetic algorithm with an adaptive local search for the MDVRP with time-windows, while the depot where the vehicle ends its route is flexible. The MDVRP with time-windows is also studied in the work Bae and Moon (2016), where a hybrid genetic algorithm is developed. An adaptive large neighbourhood search based matheuristic is proposed by Mancini (2016) for the MDVRP with Multiple Periods. Du et al. (2017) developed a fuzzy bi-level programming model for the MDVRP for hazardous materials, where the objective function is to minimise the total expected transportation risk when delivering such products. Recently, Bezerra, De Souza, and Souza (2018) and Alinaghian and Shokouhi (2018) solved the MDVRP through variants of the Variable Neighbourhood Search algorithm.

In face of the literature review, we can conclude that few mathematical programming models for the multi-depot problems have been proposed. However, exact models allow us to obtain the optimal solution, and even let us evaluate how far the obtained solution is from the optimal one in the case of it not being solved to optimality. Adding to these facts, a good mathematical programming formulation of any kind of problem is a fundamental part for the development of heuristic or meta-heuristic procedures, since it provides very useful insights regarding the problem's characteristics. Thus, the use of mathematical formulations is advantageous when compared to the straight use of heuristics. Within the MDVRP problem, as this involves the definition of service areas, which is typically a medium-term problem, the use of mathematical formulations appears suitable as they may reveal adequate in computational terms, while not consuming too many resources. Besides the previous facts, the different formulations presented in the literature for the VRP are not all extended to a multi-depot setting, where a heterogeneous vehicle fleet and routing time limits are explored. This represents an opportunity for further research, as it describes more accurately the reality of complex logistics systems.

In conclusion, the present work explores the above opportunities and proposes a MILP model for the multi-depot VRP, with heterogeneous vehicles based on the two-commodity formulation for the CVRP. A generalisation of the three-index VRP formulation for the MDVRP is also proposed and four different subtour elimination constraints are studied. Benchmark instances (original and modified) are solved by all formulations and the results are compared.

3. Alternative formulations for the MDVRP

In this section, the developed formulations are described. Firstly, the two-commodity flow formulation is presented. This formulation extends the one proposed by Baldacci, Hadjiconstantinou, and Mingozzi (2004) in several aspects: it accounts for multi-depots, a vehicle fleet with different capacities is available and introduces a route duration limit. To be able to assess this formulation quality, the traditional three-index formulation was adapted to the MDVRP problem. As the latter involves subtour elimination constraints, and in the literature there are several formulations to such constraints, four of the more common approaches are explored and comprehensively described, highlighting the multi-depot features.

3.1. Two-commodity flow formulation for the MDVRP

As stated above, the two-commodity flow formulation for the MDVRP described below is based on the two-commodity flow formulation for the CVRP, proposed by Baldacci, Hadjiconstantinou, and Mingozzi (2004). This formulation was introduced by Finke, Claus, and Gunn (1984) to solve a Travelling Salesman Problem (TSP). In their problem, two different commodities had to be delivered and collected: one unit of commodity A had to be delivered at each node and one unit of commodity B had to be collected from each node. The salesman started the tour at node 0 with n units of commodity A and 0 units of commodity B. Then he returned to node 0 with n units of commodity B and 0 units of commodity A. At each node, the salesman left one unit of A and picked up one unit of B.

Baldacci, Hadjiconstantinou, and Mingozzi (2004), based on this approach, presented a formulation for the symmetric CVRP. In their formulation, a copy of the depot is added to the original graph, which represents the node where all routes end. In opposition to the Finke, Claus, and Gunn (1984) formulation, only one commodity must be delivered according to the demand of each customer. Nonetheless, the two flow variables, y_{ij} and y_{ji}, are still present in the CVRP formulation, but with a different meaning. In fact, for Finke, Claus, and Gunn (1984), y_{ij} represents the amount of commodity A still to be delivered or, in other words, the number of nodes left to be visited (since one unit of A has to be delivered at each node, there is a one-to-one relation between nodes and product A) while y_{ji} denotes the amount of commodity B already collected or the number of nodes already visited, again given the one to one relation between nodes and product amounts. In Baldacci, Hadjiconstantinou, and Mingozzi (2004), y_{ij} denotes the vehicle load and y_{ji} represents the vehicle empty space. The flow variables define two paths for each route: the path from the real to the copy depot is defined by the flow variable representing the vehicle load (y_{ij}), while the path from the copy depot to real one is defined by the vehicle's empty space (y_{ji}). Therefore, for every route edge (i,j) one has $y_{ij} + y_{ji} = Q$ (being Q the vehicle capacity). If the edge (i,j) is part of the solution, the binary variable x_{ij} is equal to 1.

To address the multi-depot context with the two-commodity flow formulation, we extend the graph $\overline{G} = (\overline{V}, \overline{A})$ by adding the vertex subset $V_f = \{n + w + 1, \ldots, n + 2w\}$ which is a copy of the depots set. Thus $\overline{V} = V \cup V_f$, $\overline{A} = A \cup \{(i, j): i \in V_c, j \in V_f\}$ and $d_{i(j+w)} = d_{ji}$, $i \in V_c$, $j \in V_d$. Moreover, we assume a heterogeneous vehicle fleet which leads to the redefinition of vehicles' capacity Q to Q_k as the capacity of vehicle k. Different cost structures for the different types of vehicles are not considered, since only variable costs (distance minimisation) define the objective function.

To formulate the MDVRP based on the two-commodity flow formulation, the flow and binary variables presented by Baldacci, Hadjiconstantinou, and Mingozzi (2004) are redefined by adding a third index (vehicle index k), a new decision variable is needed (z_{ik}) and six new constraints are added (Constraints (8)–(13)). The three decision variables are:

- x_{ijk}, a binary variable representing the routing solution:
  1, if site j is visited immediately after site i, by vehicle k; 0, otherwise;
- y_{ijk}, a flow variable representing the vehicle load when vehicle k travels from i to j. The flow y_{jik} represents vehicle's k empty space; therefore $y_{ijk} + y_{jik} = Q_k$;
- z_{ik}, a binary variable assigning vehicles to customers:
  1, if site i is visited by vehicle k; 0, otherwise.

All routes start at one of the real depots (set V_d) and end at the corresponding copy depot (set V_f) (see Figure 1, where 11 customers have to be served from three different depots (i.e. $n = 11$, $w = 3$ and thus $V_c = \{1, 2, \ldots, 11\}$, $V_d = \{12, 13, 14\}$ and $V_f = \{15, 16, 17\}$). Each route is defined by two flow paths: one from the real to the copy depot, defined by variables y_{ijk} (representing the vehicle load);

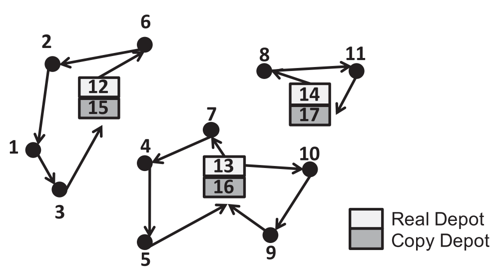

Figure 1. Illustration of the beginning and ending of routes.

and the other, the reverse path, starts at the copy depot and ends at the real one, defined by variables y_{jik} (representing the empty space on the vehicle). An illustration of these paths is represented in Figure 2.

The indices, sets and parameters included in this new formulation are the following:

Indices

i,j	node index
k	vehicle index

Sets

$\bar{V}$	the set of nodes $\bar{V} = \{1, \ldots, n+2w\}$; $\overline{V} = V_c \cup V_d \cup V_f$
V_c	the subset of customers nodes $V_c = \{1, \ldots, n\}$
V_d	the subset of real depots nodes $V_d = \{n+1, \ldots, n+w\}$
V_f	the subset of copy depots nodes $V_f = \{n+w+1, \ldots, n+2w\}$
K	the set of vehicles $K = \{1, \ldots, r\}$; $K = K_1 \cup \ldots \cup K_i$
K_i	the subset of vehicles belonging to depot i

Parameters

d_{ij}	distance between nodes i and j
r_{ij}	travelling time from node i to node j
Q_k	capacity of vehicle k
p_i	customer i demand
t_i	service duration at customer i
T	maximum time allowed for a route

The new mathematical formulation for the MDVRP is as follows:

$$Min\ F = \frac{1}{2}\sum_{i\in\bar{V}}\sum_{j\in\bar{V}}\sum_{k\in K} x_{ijk}d_{ij} \qquad (1)$$

$$\sum_{j\in\bar{V}}(y_{jik} - y_{ijk}) = 2p_i z_{ik} \quad \forall i \in V_c,\ \forall k \in K \qquad (2)$$

$$\sum_{i\in V_d}\sum_{j\in V_c}\sum_{k} y_{ijk} = \sum_{j\in V_c} p_j \qquad (3)$$

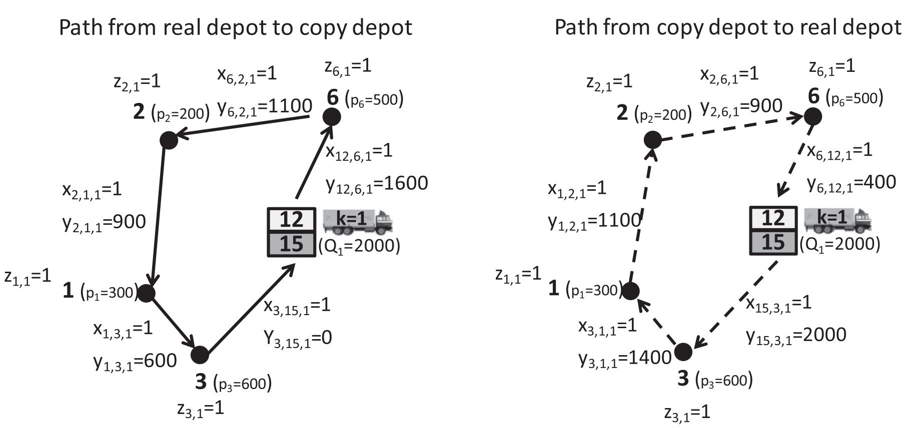

Figure 2. Illustration of the two paths defining a route solution.

$$\sum_{i\in V_d}\sum_{j\in V_c}\sum_{k} y_{jik} \leq \sum_{k} Q_k - \sum_{j\in V_c} p_j \tag{4}$$

$$\sum_{j\in V_c} y_{ijk} \leq Q_k \quad \forall i \in V_f, \forall k \in K_i \tag{5}$$

$$\sum_{i\in \bar{V}} x_{ijk} = 2z_{jk} \quad \forall j \in V_c, \forall k \in K \tag{6}$$

$$y_{ijk} + y_{jik} = Q_k x_{ijk} \quad \forall i \in \bar{V}, \forall j \in \bar{V}, \forall k \in K \tag{7}$$

$$\sum_{k\in K} z_{ik} = 1 \quad \forall i \in V_c \tag{8}$$

$$y_{ijk} \leq BigMz_{ik} \quad \forall i \in V_c, \forall j \in \bar{V}, \forall k \in K \tag{9}$$

$$\sum_{i\in V_c}\sum_{j\in \overline{V}} t_i x_{ijk} + \sum_{i\in \overline{V}}\sum_{j\in \overline{V}} r_{ij} x_{ijk} \leq 2T \quad \forall k \in K \tag{10}$$

$$\sum_{j\in V_c} x_{ijk} \leq 1 \quad \forall i \in V_d, \forall k \in K_i \tag{11}$$

$$\sum_{i\in V_c} x_{ijk} = 0 \quad \forall j \in V_f, \forall k \notin K_j \tag{12}$$

$$\sum_{j\in V_c} x_{ijk} = 0 \quad \forall i \in V_d, \forall k \notin K_i \tag{13}$$

$$y_{ijk} \geq 0, x_{ijk} \in \{0,\ 1\}, z_{ik} \in \{0,\ 1\} \quad \forall i, j \in \overline{V}, k \in K \tag{14}$$

The objective function (1) focuses on minimising the total distance travelled. Since two paths define routes, each edge of the solution is counted twice, doubling the travelled distance. Therefore, to identify the real distance, the objective function has to be divided by 2 to eliminate the distance of the second path. Constraint (2) states that the inflow minus the outflow at each customer is equal to twice the demand of each customer. Constraint (3) ensures that the total outflow of real depots is equal to the total customer demand. Regarding constraint (4), notice that the total inflow of real depots corresponds to the residual capacity of the vehicles used. Since vehicles' capacity may exceed customers' needs, leaving some vehicles unused, constraint (4) ensures that the total inflow of real depots is, at most, the residual capacity of the vehicle fleet. The total outflow of each copy depot relates to the capacity of the vehicle fleet, based at the corresponding real depot. Therefore, constraint (5) states that the outflow of each vehicle belonging to the copy depot i is less or equal to that vehicle's capacity. If a vehicle is not used, the outflow will then be zero; if a vehicle is used, the outflow will equal that vehicle's capacity. Constraint (6) guarantees that any feasible solution contains two edges incident to each customer. Constraint (7) ensures that the inflow plus the outflow of any node equals the capacity of the vehicle visiting the node. Each customer must be visited by a single vehicle (Equation (8)). Constraint (9) sets to zero the flow variable y_{ijk} if customer i is not visited by vehicle k. Constraint (10) guarantees that the duration of each route (including service and travelling time) does not exceed the maximum routing time allowed. Constraint (11) ensures that each vehicle will leave its home depot once at most. Finally, constraints (12) and (13) jointly ensure that a vehicle cannot leave and return to a depot other than its home depot (both real and copy). Variables' domains are given at constraint (14).

3.2. Three-index formulation for the MDVRP

The classical formulation for the MDVRP is the three-index formulation introduced by Golden, Magnanti, and Nguyen (1977). The decision variable is a binary variable x_{ijk} that indicates whether vehicle k travels directly from customer i to customer j. In Golden, Magnanti, and Nguyen (1977)'s formulation, the set of vehicles is not split by depots. Therefore, we have generalised their formulation to model this characteristic of the problem, rendering it equivalent to the two-commodity flow formulation presented in Section 3.1. The complete formulation is given below:

$$Min \quad \sum_{i \in V} \sum_{j \in V} \sum_{k \in K} x_{ijk} d_{ij} \tag{15}$$

$$\sum_{i \in V} \sum_{k \in K} x_{ijk} = 1 \quad \forall j \in V_c \tag{16}$$

$$\sum_{j \in V} \sum_{k \in K} x_{ijk} = 1 \quad \forall i \in V_c \tag{17}$$

$$\sum_{i \in V} x_{ihk} - \sum_{j \in V} x_{hjk} = 0 \quad \forall k \in K, \ \forall h \in V \tag{18}$$

$$\sum_{i \in V_c} \sum_{j \in V} p_i x_{ijk} \leq Q_k \quad \forall k \in K \tag{19}$$

$$\sum_{i \in V} \sum_{j \in V} t_i x_{ijk} + \sum_{i \in V} \sum_{j \in V} r_{ij} x_{ijk} \leq T \quad \forall k \in K \tag{20}$$

$$\sum_{j \in V_c} x_{ijk} \leq 1 \quad \forall k \in K_i, \forall i \in V_d \tag{21}$$

$$\sum_{i \in V_c} x_{ijk} \leq 1 \quad \forall k \in K_j, \forall j \in V_d \tag{22}$$

$$\sum_{i \in V_c} x_{ijk} = 0 \quad \forall j \in V_d, \ \forall k \notin K_j \tag{23}$$

$$\sum_{j \in V_c} x_{ijk} = 0 \quad \forall i \in V_d, \ \forall k \notin K_i \tag{24}$$

$$\text{Subtours elimination constraint} \tag{25}$$

$$x_{ijk} \in \{0, \ 1\} \quad \forall i \in V, \forall j \in V, \forall k \in K \tag{26}$$

The objective function (15) states that total distance travelled is to be minimised. Constraints (16) and (17) ensure that each customer is visited exactly once by a single vehicle. Route continuity is guaranteed by constraint (18), i.e, if a vehicle enters a site, it must exit that site. Constraint (19) ensures that the vehicle capacity is not exceeded. Similarly, constraint (20) guarantees that route duration (including service time duration and travelling time between nodes) does not exceed the maximum time allowed. Constraints (21) and (22) ensure that each vehicle will leave and return to its home depot at most once. Constraint (23) and (24) jointly ensure that a vehicle cannot leave and return to a depot other than its home depot. Constraint (25) is the subtour elimination constraint. Since in the literature some different formulations have been proposed to model this constraint, a more detailed analysis will follow below. Finally, constraint (26) sets the variables domain.

Compared to the original formulation of Golden, Magnanti, and Nguyen (1977), it can be said that Equations (15)–(20) and Equation (25) are the original ones, while Equations (21) and (22) are a generalisation, and Equations (23) and (24) are being new constraints carrying out the vehicles' partition by depots.

For the most part of the routing problems, the designed routes are Hamiltonian cycles. This means that each route defines a cycle that visits each vertex exactly once before returning to the starting vertex. The exception is the Open Vehicle Routing Problem, where the vehicles are not required to return to the depot after completing their service. Therefore, if the starting vertex corresponds to the depot, we have a feasible route. However, within this type of formulations the creation of subtours may happen. For example, if the starting vertex is one of the customers to be visited, we have a cycle called subtour, which is an infeasible route since it doesńt start and end at a depot (see Figure 3). This situation must be avoided.

In order to prevent the occurrence of these subtours, it is necessary to add extra constraints to the model. In this paper, four modelling alternatives (presented in the literature) are tested.

3.2.1. Dantzig–Fulkerson–Johnson constraint

Dantzig, Fulkerson, and Johnson (1954) formulated the subtour elimination constraint for the Travelling Salesman Problem (TSP) as:

$$\sum_{i \in S} \sum_{j \in S} x_{ij} \leq |S| - 1 \quad S \subseteq V_c, \;\; 2 \leq |S| \leq n \tag{25a}$$

Such method provides a very tight LP relaxation but, in many cases, it has the disadvantage of leading to the addition of an exponential number of constraints. However, it is often the case that not all subtour inequalities need to be added to the formulation from the beginning. They can be generated only as required by a separation algorithm: the formulation solution can start without any constraints (26a), and then if subtours occur, the corresponding subtour inequalities are added to avoid such situations.

For the MDVRP, Equation (25a) must be redefined:

$$\sum_{i \in S} \sum_{j \in S} x_{ijk} \leq |S| - 1 \quad S \subseteq V_c, \;\; 2 \leq |S| \leq n, \forall k \in K \tag{25b}$$

3.2.2. Miller–Tucker–Zemlin constraint

Miller, Tucker, and Zemlin (1960) proposed another formulation to eliminate subtours, also for the TSP:

$$u_i - u_j + n \times x_{ij} \leq n - 1 \quad 1 \leq i \neq j \leq n \tag{25c}$$

where u_i is an integer variable associated with customer i and the corresponding inequalities serve to eliminate tours that do not begin and end at a depot. The variables u_i ($i = 1, \dots, n$) represent the rank-order in which the customers are visited, with the depot being assigned a rank of zero (Sherali and Driscoll 2002). It indeed excludes subtours, as the constraint for edge (i,j) forces $u_j \geq u_i + 1$, when x_{ij} =1. The advantage of the Miller–Tucker–Zemlin (MTZ) formulation is in its small size (only n extra

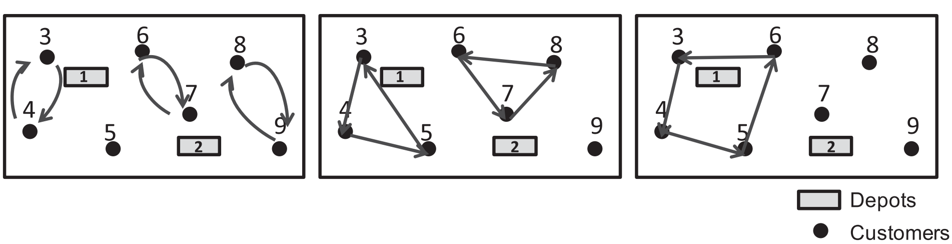

Figure 3. Subtour illustration.

variables and roughly $n^2/2$ extra constraints are needed), but the LP relaxation of the MTZ formulation is much weaker than the formulation with constraint (25a) (Pataki 2003).

For the MDVRP, Equation (25c) is redefined like follows:

$$u_i - u_j + n \times x_{ijk} \le n - 1 \quad 1 \le i \ne j \le n, \forall k \in K \tag{25d}$$

3.2.3. Transit load constraint

Kara, Laporte, and Bektas (2004) extend the MTZ subtour elimination constraint for the CVRP. Following this work, Ganesh and Narendran (2008) propose two equations to formulate the transit load constraint to eliminate subtours (constraints (25e) and (25f)). The decision variable u_i is the total demand of nodes on the route until node i (including node i).

$$u_j \ge u_i - p_j + Q(x_{ij} - 1) \quad 1 \le i \ne j \le n \tag{25e}$$

$$u_j \le u_i - p_j - Q(x_{ij} - 1) \quad 1 \le i \ne j \le n \tag{25f}$$

So, for the MDVRP, such equations result into two different options, i.e. Equation (25g) or Equations (25h) and (25i).

$$u_{ik} - u_{jk} + Q_k \times x_{ijk} \le Q_k - p_j \quad 1 \le i \ne j \le n, \ \forall k \in K \tag{25g}$$

$$u_{jk} \ge u_{ik} - p_j + Q_k(x_{ijk} - 1) \quad 1 \le i \ne j \le n, \forall k \in K \tag{25h}$$

$$u_{jk} \le u_{ik} - p_j - Q_k(x_{ijk} - 1) \quad 1 \le i \ne j \le n, \forall k \in K \tag{25i}$$

where u_{ik} is a variable decision representing the remaining load to be delivered by vehicle k when departing from node i. Notice that if arc (i,j) is visited by vehicle k, the vehicle load has to decrease by p_j.

3.2.4. Arrival time constraint

Following the approach of Kohl and Madsen (1997), one may resort to the arrival time constraints to eliminate subtours. These constraints are imposed by the problem studied by those authors, the vehicle routing problem with time windows, since it is necessary to respect customers' time-windows.

For the MDVRP studied in this paper, there are no time-windows, but all customers have to be served within a workday. Therefore, constraint (25j) assures subtour elimination:

$$B_{jk} \ge B_{ik} + t_i + r_{ij} - M(1 - x_{ijk}) \quad 1 \le i \ne j \le n, \ \forall k \in K \tag{25j}$$

with B_{jk} the variable modelling the arrival time of vehicle k at customer j and constant $M \ge T$.

Summarising, and based on the above options, five different formulations to prevent subtours will be tested in the three-index formulation:

- Dantzig–Fulkerson–Johnson (DFJ) constraints (Equation (25b))
- Dantzig–Fulkerson–Johnson constraints, with a separation algorithm (cutting-plane fashion)
- Miller–Tucker–Zemlin constraints (Equation (25d))
- Transit load constraints (Equation (25g) or (25h) and (25i))
- Arrival time constraints (Equation (25j))

5. Test results

In this section, we evaluate the alternative formulations for the MDVRP by applying it to a set of test instances and comparing some performances measurements (i.e. the solution quality and the computational time). The original 33 benchmarking instances available for the MDVRP (P01–P23 and PR01–PR10, see Cordeau, Gendreau, and Laporte (1997)) are medium to large size scale, and the exact approaches explored in this work are only able to solve small to medium size scale instances. Thus, only medium size instances were selected, the ones with $n \leq 100$ customers: P01–P07, P12–P14, PR01, PR02, PR07 (13 instances in total). To increase the set of instances and given that all PR instances have both capacity and duration constraints, we modified the PR instances to obtain small and medium size instances. For that, we select the first 25 and 40 customers from the PR instances, maintaining the original number and location of the depots. This strategy was inspired by the work of Muter, Cordeau, and Laporte (2014). The number of vehicles was also adjusted to one vehicle for depot for the instances with 25 customers, and 2 vehicles for depot for the instances with 40 customers (except for PR01 and PR02, where only one vehicle belongs to each depot in the original data). Therefore, our test bed has 33 instances, where 13 are the original benchmark instances for the MDVRP available in the literature (Cordeau, Gendreau, and Laporte 1997) and 20 are a modified version of the original ones. These instances' structure is shown in Table 1, as well as, the Best Known Solution in the literature.

As one of this paper's objectives is to assess how large an instance of the MDVRP can be solved by a commercial solver, namely the CPLEX by ILOG, all instances were solved with the models implemented in GAMS software using the branch-and-bound method with the CPLEX Optimiser 12.7.0. An Intel Xeon CPU X5680 @ 3.33 GHz is used. The branch-and-bound computation time is limited to 4 h (14,400 s) for the modified instances and 8 h (28,800 s) for the original ones.

Additionally, we want to compare the performance of the developed two-commodity flow formulation with the three-index formulation to solve the MDVRP, and within the three-

Table 1. Structure of the test instances.

Instance	Literature instance	No. of depots (w)	No. of customers (n)	No. of vehicles	Best known solution
PR01_25	Modified	4	25	4	–
PR02_25	Modified	4	25	4	–
PR03_25	Modified	4	25	4	–
PR04_25	Modified	4	25	4	–
PR05_25	Modified	4	25	4	–
PR06_25	Modified	4	25	4	–
PR07_25	Modified	6	25	6	–
PR08_25	Modified	6	25	6	–
PR09_25	Modified	6	25	6	–
PR010_25	Modified	6	25	6	–
PR01_40	Modified	4	40	4	–
PR02_40	Modified	4	40	8	–
PR03_40	Modified	4	40	8	–
PR04_40	Modified	4	40	8	–
PR05_40	Modified	4	40	8	–
PR06_40	Modified	4	40	8	–
PR07_40	Modified	6	40	6	–
PR08_40	Modified	6	40	12	–
PR09_40	Modified	6	40	12	–
PR010_40	Modified	6	40	12	–
P01	Original	4	50	16	576.87
P02	Original	4	50	8	473.53
P03	Original	5	75	15	641.19
P04	Original	2	100	16	1001.04
P05	Original	2	100	10	750.03
P06	Original	3	100	18	876.5
P07	Original	4	100	16	881.97
P12	Original	2	80	10	1318.95
P13	Original	2	80	10	1318.95
P14	Original	2	80	10	1360.12
PR01	Original	4	48	4	861.32
PR02	Original	4	96	8	1307.34
PR07	Original	6	72	6	1089.56

index formulation, we also aim to assess the performance of different formulations to prevent subtours (DFJ constraints; DFJ with separation algorithm; MTZ constraints; transit load constraints; and arrival time constraints). As mentioned in the previous section, for the transit load formulation different inequalities are available in the literature. Therefore, we have performed some preliminary simulations to test the performance of each constraint and Equation (25g) overcomes the performance of Equations (25h) and (25i). Based on this, the following results are obtained by considering Equation (25g) as the transit load constraint to prevent subtours.

Table 2 compares the alternative formulations addressed in terms of their size: the total number of constraints and variables. It can be seen that the MTZ formulation has the fewest variables and constraints, whereas the flow formulation has the most. The transit load and arrival time constraints have an equal number of variables and constraints. The DFJ formulation is not shown in Table 2 because the GAMS software was not able to generate the model, even for the smaller instances with 25 customers only.

Table 3 reports the objective function value (OFV), the total computing time (CPU), the percentage ratio between the lower bound obtained at the root node and the objective function value (%LB) and the gap between the integer solution and the lower bound found by CPLEX after the total computing time (GAP) for each instance and formulation studied.

It can be seen that the four formulations – MTZ, transit load, arrival time and two-commodity flow formulations – found the optimal solution for all instances with 25 customers (PR01_25 to PR10_25). However, the two-commodity flow formulation was able to prove optimality for all instances and it consumed consistently less computational time than all other formulations. On average, optimality has been proven in 129 s using the two-commodity flow formulation, and in 1065 s using the three-index formulation. In one instance only, PR08_25, the transit load was not able to prove optimality within the time limit.

For the instances with 40 customers (PR01_40 to PR10_40), the two-commodity flow formulation proved optimality in 5 instances (PR01_40, PR04_40 to PR07_40) and lower gaps were obtained for the remaining 5 instances (gaps lower or equal than 5%). The three-index formulations reached the 4-h time limit providing worse solutions, with high gaps in almost all 40-customers instances. Nonetheless, for instances PR01_40 and PR06_40, the three-index formulations also found the optimal solution but were not able to prove optimality within the 4-h time limit.

Table 2: Number of variables (#Var.) and constraints (#Con.) of the test instances.

			Three-index formulation					
	Flow formulation		MTZ		Transit load		Arrival time	
Instance	#Var.	#Con.	#Var.	#Con.	#Var.	#Con.	#Var.	#Con.
PR01_25 to PR06_25	8681	7704	3374	2707	3465	3107	3465	3107
PR07_25 to PR10_25	16357	13834	5756	4071	5917	4971	5917	4971
PR01_40 to PR06_40	36801	33875	15497	13313	15809	14593	15809	14593
PR07_40 to PR10_40	64753	57535	25361	20001	25873	22881	25873	22881
P01	353281	338589	162007	150093	163721	155253	163721	155253
P02	84601	78741	80065	74801	80769	77873	80769	77873
P03	215476	204153	96001	85891	97126	91516	–	–
P04	344449	339591	166533	161913	168065	165113	–	–
P05	215281	212283	104121	101271	105041	103271	–	–
P06	402589	393205	191009	182181	192763	187581	–	–
P07	371521	359591	173093	162009	174657	168409	–	–
P12	140281	137863	67301	65031	68041	66631	–	–
P13	140281	137873	67301	65041	68041	66641	68041	66641
P14	140281	137873	67301	65041	68041	66641	68041	66641
PR01	25057	23367	10849	9561	11009	10329	11009	10329
PR02	172993	166539	80065	74801	80769	77873	80769	77873
PR07	84601	78741	36541	31801	36937	34393	36937	34393

'–' the test instance does not have an arrival time.

Table 3. Test instances results.

	OFV	CPU (s)	%LB	GAP (%)
PR01_25				
Three-index formulation				
1) MTZ formulation	590.35	142	73%	0%
2) Transit load formulation	590.35	35	73%	0%
3) Arrival time formulation	590.35	183	73%	0%
Two-commodity flow formulation	590.35	21	79%	0%
PR02_25				
Three-index formulation				
1) MTZ formulation	716.38	2617	59%	0%
2) Transit load formulation	716.38	2096	59%	0%
3) Arrival time formulation	716.38	6228	59%	0%
Two-commodity flow formulation	716.38	246	66%	0%
PR03_25				
Three-index formulation				
1) MTZ formulation	575.40	329	81%	0%
2) Transit load formulation	575.40	198	81%	0%
3) Arrival time formulation	575.40	568	81%	0%
Two-commodity flow formulation	575.40	97	85%	0%
PR04_25				
Three-index formulation	703.52	107	83%	
1) MTZ formulation	703.52	85	83%	0%
2) Transit load formulation	703.52	241	83%	0%
3) Arrival time formulation	703.52	18	87%	0%
Two-commodity flow formulation				0%
PR05_25				
Three-index formulation				
1) MTZ formulation	481.54	319	78%	0%
2) Transit load formulation	481.54	1120	78%	0%
3) Arrival time formulation	481.54	177	78%	0%
Two-commodity flow formulation	481.54	27	85%	0%
PR06_25				
Three-index formulation				
1) MTZ formulation	659.56	470	79%	0%
2) Transit load formulation	659.56	488	79%	0%
3) Arrival time formulation	659.56	354	79%	0%
Two-commodity flow formulation	659.56	63	81%	0%
PR07_25				
Three-index formulation				
1) MTZ formulation	684.01	148	80%	0%
2) Transit load formulation	684.01	203	80%	0%
3) Arrival time formulation	684.01	160	80%	0%
Two-commodity flow formulation	684.01	43	84%	0%
PR08_25				
Three-index formulation				
1) MTZ formulation	612.48	2164	68%	0%
2) Transit load formulation	612.48	14,400	68%	6%
3) Arrival time formulation	612.48	8014	68%	0%
Two-commodity flow formulation	612.48	555	72%	0%
PR09_25				
Three-index formulation				
1) MTZ Formulation	707.12	589	72%	0%
2) Transit Load Formulation	707.12	3387	72%	0%
3) Arrival Time Formulation	707.12	288	72%	0%
Two-Commodity Flow Formulation	707.12	207	78%	0%
PR10_25				
Three-index formulation				
1) MTZ formulation	518.95	45	72%	0%
2) Transit load formulation	518.95	88	72%	0%
3) Arrival time formulation	518.95	56	72%	0%
Two-commodity flow formulation	518.95	8	78%	0%
PR01_40				
Three-index formulation				
1) MTZ formulation	726.82	14,400	83%	6%

(*Continued*)

Table 3. Continued.

	OFV	CPU (s)	%LB	GAP (%)
2) Transit load formulation	726.82	14,400	83%	5%
3) Arrival time formulation	726.82	14,400	83%	25%
Two-commodity flow formulation	726.82	74	88%	0%
PR02_40				
Three-index formulation				
1) MTZ formulation	1162.29	14,400	55%	30%
2) Transit load formulation	1044.69	14,400	61%	19%
3) Arrival time formulation	964.56	14,400	67%	11%
Two-commodity flow formulation	908.73	14,400	78%	5%
PR03_40				
Three-index formulation				
1) MTZ formulation	884.24	14,400	80%	13%
2) Transit load formulation	909.19	14,400	78%	18%
3) Arrival time formulation	951.92	14,400	74%	15%
Two-commodity flow formulation	882.10	14,400	83%	2%
PR04_40				
Three-index formulation				
1) MTZ formulation	884.27	14,400	71%	22%
2) Transit load formulation	809.64	14,400	77%	6%
3) Arrival time formulation	809.64	14,400	77%	6%
Two-commodity flow formulation	809.64	4716	83%	0%
PR05_40				
Three-index formulation				
1) MTZ formulation	674.89	14,400	79%	11%
2) Transit load formulation	674.89	14,400	79%	9%
3) Arrival time formulation	1001.34	10,561[1]	53%	40%
Two-commodity flow formulation	673.45	32	88%	0%
PR06_40				
Three-index formulation				
1) MTZ formulation	743.47	14,400	80%	16%
2) Transit load formulation	743.47	14,400	80%	7%
3) Arrival time formulation	743.47	14,400	80%	14%
Two-commodity flow formulation	743.47	330	85%	0%
PR07_40				
Three-index formulation				
1) MTZ formulation	848.08	14,400	78%	15%
2) Transit load formulation	843.19	14,400	78%	16%
3) Arrival time formulation	853.52	14,400	77%	15%
Two-commodity flow formulation	839.93	688	83%	0%
PR08_40				
Three-index formulation				
1) MTZ formulation	780.14	14,400	62%	22%
2) Transit load formulation	835.63	14,400	58%	26%
3) Arrival time formulation	765.51	14,400	63%	19%
Two-commodity flow formulation	755.46	14,400	71%	5%
PR09_40				
Three-index formulation				
1) MTZ formulation	–	–	–	–
2) Transit load formulation	969.01	14,400	66%	31%
3) Arrival time formulation	966.29	14,400	67%	31%
Two-commodity flow formulation	864.03	14,400	81%	5%
PR10_40				
Three-index formulation				
1) MTZ formulation	843.39	11,501[1]	67%	26%
2) Transit load formulation	1053.86	11,033[1]	54%	40%
3) Arrival time formulation	787.79	14,400	72%	20%
Two-commodity flow formulation	738.22	14,400	83%	4%
P01				
Three-index formulation				
1) MTZ formulation	576.87	28,800	65%	32%
2) Transit load formulation	822.90	28,800	45%	52%
3) Arrival time formulation	NA	NA	NA	NA
Two-commodity flow formulation	576.87	28,800	85%	9%

(*Continued*)

Table 3. Continued.

	OFV	CPU (s)	%LB	GAP (%)
P02				
Three-index formulation				
1) MTZ formulation	492.89	28,800	76%	15%
2) Transit load formulation	492.70	28,800	76%	14%
3) Arrival time formulation	NA	NA	NA	NA
Two-commodity flow formulation	473.53	28,800	88%	4%
P03				
Three-index formulation				
1) MTZ formulation	–	–	–	–
2) Transit load formulation	–	–	–	–
3) Arrival time formulation	NA	NA	NA	NA
Two-commodity flow formulation	641.19	28,800	88%	6%
P04 to P07				
Three-index formulation				
1) MTZ formulation	–	–	–	–
2) Transit load formulation	–	–	–	–
3) Arrival time formulation	–	–	–	–
Two-commodity flow formulation	–	–	–	–
P12				
Three-index formulation				
1) MTZ formulation	–	–	–	–
2) Transit load formulation	–	–	–	–
3) Arrival time formulation	NA	NA	NA	NA
Two-commodity flow formulation	1318.95	28,800	80%	5%
P13				
Three-index formulation				
1) MTZ formulation	–	–	–	–
2) Transit load formulation	–	–	–	–
3) Arrival time formulation	–	–	–	–
Two-commodity flow formulation	1318.95	28,800	80%	12%
P14				
Three-index formulation				
1) MTZ formulation	–	–	–	–
2) Transit load formulation	–	–	–	–
3) Arrival time formulation	–	–	–	–
Two-commodity flow formulation	1360.12	28,800	78%	15%
PR01				
Three-index formulation				
1) MTZ formulation	–	–	–	–
2) Transit load formulation	1143.30	28,800	62%	28%
3) Arrival time formulation	–	–	–	–
Two-commodity flow formulation	861.32	86	88%	0%
PR02				
Three-index formulation				
1) MTZ formulation	–	–	–	–
2) Transit load formulation	–	–	–	–
3) Arrival time formulation	–	–	–	–
Two-commodity flow formulation	–	–	–	–
PR07				
Three-index formulation				
1) MTZ formulation	–	–	–	–
2) Transit load formulation	–	–	–	–
3) Arrival time formulation	–	–	–	–
Two-commodity flow formulation	1089.56	28,800	86%	3%

'–' implies that no integer solution was found by CPLEX within the 8-h limit.
'NA' the test instance does not have arrival time.
'1' Out of memory.

For the original literature instances, the two-commodity flow formulation was able to prove the optimality of instance PR01 in 86 s and, although without proving their optimality it found the best known solutions for the remaining instances. The solutions and gaps achieved by the three-index formulation in the literature instances are worse than the ones found by the flow formulation.

For instances P03, P12–P14 and PR07, only the two-commodity flow formulation was capable to find an integer solution. However, for the instances with 100 customers (P04–P07 and PR02), the flow formulation did not provide any integer solution.

Analysing the percentage ratio between the lower bound obtained at the root node and the optimal value, we conclude that the two-commodity flow formulation provides a tighter LP relaxation than the other formulations.

Comparing the subtour elimination constraints in the three-index formulation, no formulation presented consistently better results than any others.

6. Conclusions

In this paper, the MDVRP problem has been studied and a new two-commodity flow formulation was proposed. Given a heterogeneous vehicle fleet based at each depot, delivery routes to serve a set of customers with known demands are defined, assuming that the vehicle capacity and the total duration of a route cannot be exceeded and that the total distance travelled must be minimised.

Furthermore, a more traditional formulation – the three-index formulation – was generalised to the same MDVRP problem, and four alternative formulations to eliminate subtours were studied: Dantzig–Fulkerson–Johnson constraint, Miller–Tucker–Zemlin constraint, transit load constraint and arrival time constraint.

A comparison between the proposed two-commodity flow formulation and the adapted three-index formulation was performed using a commercial solver (e.g. CPLEX). In all tested instances, the flow formulation showed a better performance achieving lower gaps with smaller computational time. This formulation also provided a tighter LP relaxation than the three-index formulation developed. However, for the larger instances, the two-commodity flow formulation was not able to prove solution optimality within the 8-h CPU limit and, in problems with more than 100 nodes, it failed to report any integer solution.

Additionally, this work also compared alternative constraints formulations to prevent subtours within the three-index formulation. The tests showed that an exponential number of constraints generated by the Dantzig–Fulkerson–Johnson formulation prevent the solver to find an integer solution within the preset time limit. For the remaining subtours elimination constraints, the results obtained were not conclusive regarding which type of formulation performs better.

As a final conclusion, it can be said that in this paper a more efficient formulation than the traditional three-index formulation was proposed to solve the MDVRP problem. As future work, it may be an option the study of oriented solution procedures using the proposed formulation, as well as alternative solution techniques, such as a decomposition approach and/or the development of valid inequalities. We expect that better solutions will be obtained, as well as an improvement on the computational performance.

Disclosure statement

No potential conflict of interest was reported by the authors.

ORCID

Tânia Rodrigues Pereira Ramos http://orcid.org/0000-0002-2321-2431

References

Alinaghian, M., and N. Shokouhi. 2018. "Multi-depot Multi-compartment Vehicle Routing Problem, Solved by Ahybrid Adaptive Large Neighborhood Search." *Omega* 76: 85–99.

Atamturk, A., and M. W. P. Savelsbergh. 2005. "Integer-programming Software Systems." *Annals of Operations Research* 140: 67–124.

Bae, H., and I. Moon. 2016. "Multi-depot Vehicle Routing Problem with Time Windows Considering Delivery and Installation Vehicles." *Applied Mathematical Modelling* 40: 6536–6549.
Baldacci, R., N. Christofides, and A. Mingozzi. 2008. "An Exact Algorithm for the Vehicle Routing Problem Based on the set Partitioning Formulation with Additional Cuts." *Mathematical Programming* 115: 351–385.
Baldacci, R., E. Hadjiconstantinou, and A. Mingozzi. 2004. "An Exact Algorithm for the Capacitated Vehicle Routing Problem Based on a Two-commodity Network Flow Formulation." *Operations Research* 52: 723–738.
Baldacci, R., and A. Mingozzi. 2009. "A Unified Exact Method for Solving Different Classes of Vehicle Routing Problems." *Mathematical Programming* 120: 347–380.
Baldacci, R., P. Toth, and D. Vigo. 2007. "Recent Advances in Vehicle Routing Exact Algorithms." *4or-a Quarterly Journal of Operations Research* 5: 269–298.
Balinski, M. L., and R. E. Quandt. 1964. "On Integer-Program for Delivery Problem." *Operations Research* 12: 300-&.
Bettinelli, A., A. Ceselli, and G. Righini. 2011. "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: 723–740.
Bezerra, S. N., S. R. De Souza, and M. J. F. Souza. 2018. "A GVNS Algorithm for Solving the Multi-depot Vehicle Routing Problem." *Electronic Notes in Discrete Mathematics* 66: 167–174.
Braekers, K., K. Ramaekers, and I. V. Nieuwenh. 2016. "The Vehicle Routing Problem: State of the Art Classification and Review." *Computers & Industrial Engineering* 99: 300–313.
Chao, I., B. L. Golden, and E. Wasil. 1993. "A new Heuristic for the Multi-depot Vehicle Routing Problem Taht Improves upon Best-known Solutions." *American Journal of Mathematical and Management Sciences* 13: 371–406.
Chávez, J. J. S., J. W. Escobar, and M. G. Echeverri. 2016. "A Multi-objective Pareto and Colony Algorithm for the Multi-depot Vehicle Routing Problem with Backhauls." *International Journal of Industrial Engineering Computations* 7: 35–48.
Contardo, C., and R. Martinelli. 2014. "A new Exact Algorithm for the Multi-depot Vehicle Routing Problem Under Capacity and Route Length Constraints." *Discrete Optimization* 12: 129–146.
Cordeau, J. F., M. Gendreau, and G. Laporte. 1997. "A Tabu Search Heuristic for Periodic and Multi-depot Vehicle Routing Problems." *Networks* 30: 105–119.
Crevier, B., J. F. Cordeau, and G. Laporte. 2007. "The Multi-depot Vehicle Routing Problem with Inter-depot Routes." *European Journal of Operational Research* 176: 756–773.
Dantzig, G., R. Fulkerson, and S. Johnson. 1954. "Solution of a Large-scale Traveling-Salesman Problem." *Journal of the Operations Research Society of America* 2: 393–410.
Dondo, R. G. D. R. G., and J. Cerda. 2009. "A Hybrid Local Improvement Algorithm for Large-scale Multi-depot Vehicle Routing Problems with Time Windows." *Computers & Chemical Engineering* 33: 513–530.
Du, J., X. Li, L. Yu, R. Dan, and J. Zou. 2017. "Multi-depot Vehicle Routing Problem for Hazardous Materials Transportation: A Fuzzy Bilevel Programming." *Information Sciences* 399: 201–218.
Dueck, G. 1993. "New Optimization Heuristics – the Great Deluge Algorithm and the Record-to-record Travel." *Journal of Computational Physics* 104: 86–92.
Escobar, J. W., R. Linfati, P. Toth, and M. G. Baldoquin. 2014. "A Hybrid Granular Tabu Search Algorithm for the Multi-depot Vehicle Routing Problem." *Journal of Heuristics* 20: 483–509.
Finke, G., A. Claus, and E. Gunn. 1984. "A two-Commodity Network flow Approach to the Traveling Salesman Problem." *Congressus Numerantium* 41: 167–178.
Fukasawa, R., H. Longo, J. Lysgaard, M. P. De Aragao, M. Reis, E. Uchoa, and R. F. Werneck. 2006. "Robust Branch-and-cut-and-price for the Capacitated Vehicle Routing Problem." *Mathematical Programming* 106: 491–511.
Ganesh, K., and T. T. Narendran. 2008. "TASTE: A Two-phase Heuristic to Solve a Routing Problem with Simultaneous Delivery and Pick-up." *International Journal of Advanced Manufacturing Technology* 37: 1221–1231.
Gillet, B., and J. Johnson. 1976. "Multi-terminal Vehicle Dispatching Algorithm." *Omega* 4: 711–718.
Golden, B. L., T. L. Magnanti, and H. Q. Nguyen. 1977. "Implementing Vehicle Routing Algorithms." *Networks* 7: 113–148.
Gulczynski, D., B. Golden, and E. Wasil. 2011. "The Multi-depot Split Delivery Vehicle Routing Problem: An Integer Programming-Based Heuristic, New Test Problems, and Computational Results." *Computers & Industrial Engineering* 61: 794–804.
Ho, W., G. T. S. Ho, P. Ji, and H. C. W. Lau. 2008. "A Hybrid Genetic Algorithm for the Multi-depot Vehicle Routing Problem." *Engineering Applications of Artificial Intelligence* 21: 548–557.
Kara, I., G. Laporte, and T. Bektas. 2004. "A Note on the Lifted Miller-Tucker-Zemlin Subtour Elimination Constraints for the Capacitated Vehicle Routing Problem." *European Journal of Operational Research* 158: 793–795.
Kohl, N., and O. B. G. Madsen. 1997. "An Optimization Algorithm for the Vehicle Routing Problem with Time Windows Based on Lagrangian Relaxation." *Operations Research* 45: 395–406.
Laporte, G., Y. Nobert, and D. Arpin. 1984. "Optimal Solutions to Capacitated Multi-depot Vehicle Routing Problems." *Congressus Numerantium* 44: 283–292.
Laporte, G., Y. Nobert, and M. Desrochers. 1985. "Optimal Routing Under Capacity and Distance Restrictions." *Operations Research* 33: 1050–1073.

Laporte, G., Y. Nobert, and S. Taillefer. 1988. “Solving a Family of Multi-depot Vehicle-routing and Location-routing Problems.” *Transportation Science* 22: 161–172.
Li, J., Y. Li, and P. M. Pardalos. 2016. “Multi-depot Vehicle Routing Problem with Time Windows Under Shared Depot Resources.” *Journal of Combinatorial Optimization* 31: 515–532.
Lim, A., and F. Wang. 2005. “Multi-depot Vehicle Routing Problem: A one-Stage Approach.” *Ieee Transactions on Automation Science and Engineering* 2: 397–402.
Mancini, S. 2016. 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* 70: 100–112.
Miller, C. E., A. W. Tucker, and R. A. Zemlin. 1960. “Integer Programming Formulation of Traveling Salesman Problems.” *Journal of the Acm* 7: 326–329.
Montoya-Torres, J. R., J. L. Franco, S. N. Isaza, H. F. Jimenez, and N. Herazo-Padilla. 2015. “A Literature Review on the Vehicle Routing Problem with Multiple Depots.” *Computers & Industrial Engineering* 79: 115–129.
Muter, I., J. F. Cordeau, and G. Laporte. 2014. “A Branch-and-price Algorithm for the Multidepot Vehicle Routing Problem with Interdepot Routes.” *Transportation Science* 48: 425–441.
Pataki, G. 2003. “Teaching Integer Programming Formulations Using the Traveling Salesman Problem.” *Siam Review* 45: 116–123.
Renaud, J., G. Laporte, and F. F. Boctor. 1996. “A Tabu Search Heuristic for the Multi-Depot Vehicle Routing Problem.” *Computers & Operations Research* 23: 229–235.
Salhi, S., A. Imran, and N. A. Wassan. 2014. “The Multi-depot Vehicle Routing Problem with Heterogeneous Vehicle Fleet: Formulation and a Variable Neighborhood Search Implementation.” *Computers & Operations Research* 52: 315–325.
Salhi, S., and M. Sari. 1997. “A Multi-level Composite Heuristic for the Multi-depot Vehicle Fleet mix Problem.” *European Journal of Operational Research* 103: 95–112.
Sherali, H. D., and P. J. Driscoll. 2002. “On Tightening the Relaxations of Miller-Tucker-Zemlin Formulations for Asymmetric Traveling Salesman Problems.” *Operations Research* 50: 656–669.
Tillman, F. A., and T. M. Cain. 1972. “Upperbound Algorithem for Single and Multiple Terminal Delivery Problem.” *Management Science Series a-Theory* 18: 664–682.
Vidal, T., T. G. Crainic, M. Gendreau, N. Lahrichi, and W. Rei. 2012. “A Hybrid Genetic Algorithm for Multidepot and Periodic Vehicle Routing Problems.” *Operations Research* 60: 611–624.
Yu, B., N. Ma, W. Cai, T. Li, X. Yuan, and B. Yao. 2013. “Improved Ant Colony Optimisation for the Dynamic Multi-depot Vehicle Routing Problem.” *International Journal of Logistics Research and Applications* 16: 144–157.
Yucenur, G. N., and N. C. Demirel. 2011. “A New Geometric Shape-based Genetic Clustering Algorithm for the Multi-depot Vehicle Routing Problem.” *Expert Systems with Applications* 38: 11859–11865.
Zhang, Y., M. Qi, L. Miao, and G. Wu. 2015. “A Generalized Multi-depot Vehicle Routing Problem with Replenishment Based on LocalSolver.” *International Journal of Industrial Engineering Computations* 6: 81–98.