Submitted to *Transportation Science*
manuscript TS-2018-0173

A Benders Decomposition Approach for the Multi-Vehicle Production Routing Problem with Order-up-to-level Policy

Zhenzhen Zhang
Department of Industrial Systems Engineering and Management, National University of Singapore, Singapore, zhenzhenzhang222@gmail.com

Zhixing Luo
School of Management and Engineering, Nanjing University, Nanjing 210093, P.R. China, luozx.hkphd@gmail.com
Department of Industrial Systems Engineering and Management, National University of Singapore, Singapore

Roberto Baldacci
Department of Electrical, Electronic, and Information Engineering "Guglielmo Marconi", University of Bologna, Via Venezia 52, Cesena 47521, Italy, r.baldacci@unibo.it

Andrew Lim
Department of Industrial Systems Engineering and Management, National University of Singapore, Singapore, isealim@nus.edu.sg

The Production Routing Problem (PRP) arises in the applications of integrated supply chain which jointly optimizes the production, inventory, distribution, and routing decisions. The literature on this problem is quite rare due to its complexity. In this paper, we consider the multi-vehicle PRP (MVPRP) with Order-Up-to-level inventory replenishment policy, where every time a customer is visited, the quantity delivered is such that the maximum inventory level is reached. We propose an exact Benders' decomposition approach to solve the MVPRP, which decomposes the problem as a master problem and a slave problem. The master problem decides whether to produce the product, the quantity to be produced, and the customers to be replenished for every period of the planning horizon. The resulting slave problem decomposes into a Capacitated Vehicle Routing Problem for each period of the planning horizon where each problem is solved using an exact algorithm based on the set partitioning model, and the identified feasibility and optimality cuts are added to the master problem to guide the solution process. Valid inequalities and initial optimality cuts are used to strengthen the LP-relaxation of the master formulation. The exact method is tested on MVPRP instances and on instances of the multi-vehicle Vendor-Managed Inventory Routing Problem, a special case of the MVPRP, and the good performance of the proposed approach is demonstrated.

Key words: production routing problem; Logic Benders' decomposition; set partitioning model
History: Received: June 24, 2018; Revised: May 2, 2019; Accepted: July 19, 2019

1. Introduction

Inventory routing problems are among the most important and challenging extensions of Vehicle Routing Problems (VRPs) (Toth and Vigo 2014). In its basic version, the Inventory Routing Problem (IRP) is concerned with the distribution of a single product from a single facility to a set of customers over a given planning horizon. Customers consume the product at a given rate and can maintain an inventory of the product up to a specific level. A fleet of identical vehicles is available for the distribution of the product. The objective is to minimize the total distribution cost, computed as the sum of the route costs and inventory holding costs, without causing stockouts at any of the customers. In the IRP, inventory control and routing decisions have to be made simultaneously.

The Production Routing Problem (PRP) generalizes the IRP by taking the production decisions into account. The PRP considers various decisions in supply chain simultaneously, including the production, inventory management, and routing decisions. Generally speaking, over a multi-period horizon, these integrated supply chain problems require to determine the period of production and visit of each customer, the corresponding quantities of production to be produced and delivered to the customers, and the detailed routing plan of the vehicles. The aim is to minimize the total production, inventory and distribution costs.

The PRP is very difficult to solve in practice and finds a very large number of applications, e.g., in so-called Vendor-Managed Inventory (VMI) systems, in which a supplier manages the inventory replenishment of its customers (retailers). Therefore, the supplier has to decide the periods when to replenish the customers over a given planning horizon, the routes to perform, and the quantities to deliver at each visit to avoid stockouts at the customers. The application of vendor managed resupply principles creates advantages for both supplier and customers. The vendor saves on distribution costs by being able to better coordinate deliveries to different customers. Customers may receive incentives and all save time and effort on inventory management.

VMI systems were first introduced in the literature through applications in the distribution of liquid air products, but many different industries are now implementing such systems or are exploring their use. These include the automotive, electronics assembly, and chemicals industries, vending machines for juice or foods, chain stores, maritime logistics, among others (Andersson et al. 2010, Coelho et al. 2014). The number of applications is increasing, along with the need for approaches to the PRP that can handle the additional constraints and complexities found in practical contexts.

1.1. Literature review

The IRP was introduced by Campbell et al. (2002) in the context of the distribution of liquid air products. The authors described a two-phase heuristic approach based on decomposing the set of decisions into the creation of a delivery schedule, followed by the construction of a set of delivery routes. This solution methodology was designed for the solution of large-scale, real-life instances.

There is a growing number of references to inventory routing and related problems in the literature. Surveys on IRPs can be found in Bertazzi et al. (2008) and Andersson et al. (2010). The more recent survey papers of Coelho et al. (2014) and Adulyasak et al. (2015) provide a comprehensive review of the IRP literature, based on a new classification of the problems. IRPs are categorized with respect to their structural variants and with respect to the availability of information on customer demand.

An important variant of the IRP arises when both inventory control at the depot and inventory costs are considered. This problem is often called the Vendor-Managed Inventory Routing Problem (VMIRP). The PRP further generalises the VMIRP by considering, in addition to inventory control, production lot-sizing decisions at the depot.

Three main replenishment policies for the customers have been considered in the inventory routing literature (Archetti et al. 2007). In the first policy, called *order-up-to-level* (OU) policy, every time a customer is visited, the quantity delivered is such that the maximum inventory level of the customer is reached. The second policy, called *maximum-level* (ML) policy, relaxes the OU policy by allowing the quantity delivered to a customer to be any value such that the resulting inventory level is between the current level and the maximum level. The third policy, called *replenishment* (RP) policy, further relaxes the ML policy by relaxing the constraint on the maximum inventory level and by allowing the delivered quantity to be any positive value.

The single-vehicle VMIRP with the OU policy was introduced by Bertazzi et al. (2002), who proposed a heuristic algorithm for its solution. Archetti et al. (2007) developed a branch-and-cut approach for the VMIRP with a single vehicle and the three different replenishment policies described above. They could solve instances with up to 45 customers and three periods and up to 30 customers and six periods to optimality within two hours for the VMIRP with the OU and ML policies, respectively. Another branch-and-cut algorithm, based on a stronger mathematical formulation, was introduced by Solyalı and Süral (2011) who were able to solve to optimality instances with up to 60 customers and three periods or 15 customers and 12 periods. Their formulation relies on the shortest path representation of the lot sizing problem, where decision variables indicate time intervals between successive deliveries. Avella et al. (2015) took advantage of the special structure of test instances and developed tighter reformulations for the single-vehicle IRP with both OU and ML policies. The authors reported computational results on benchmark instances with 50 customers and six periods. Heuristic algorithms for the single-vehicle VMIRP have been investigated by Archetti et al. (2012).

Coelho and Laporte (2013) introduced a branch-and-cut algorithm for the multi-vehicle variant of the VMIRP. Their algorithm uses an extension of the Archetti et al. (2007) formulation and it could solve some instances with up to 45 customers, three periods and three vehicles. The results of Coelho and Laporte (2013) were further improved by Coelho and Laporte (2014) by introducing new valid inequalities based on the relation between demand and available capacities. Exact algorithms for the multi-vehicle VMIRP have also been proposed by Avella et al. (2018) and Desaulniers et al. (2016). Avella et al. (2018) described IRP reformulations under the ML replenishment policy, derived from a single-period substructure, and defined a generic family of valid inequalities. The authors described a branch-and-cut algorithm and reported computational results for the benchmark instances with 50 customers and three periods and 30 customers and six periods. Desaulniers et al. (2016) described a branch-price-and-cut algorithm based on an innovative mathematical formulation for the problem under the ML policy, tightened with the inclusion of known and new families of valid inequalities. The authors reported computational experiments on a set of 640 benchmark instances involving between two and five vehicles, showing that their branch-price-and-cut algorithm outperforms the branch-and-cut algorithm of Coelho and Laporte (2014) on the instances with four and five vehicles.

The literature on PRP problem is rather limited. Archetti et al. (2011) adapted the branch-and-cut method of Archetti et al. (2007) to solve the single-vehicle PRP under ML policy and reported computational experiments on small size instances involving 14 customers. Exact algorithms for the multi-vehicle PRP (MVPRP) have been proposed by Bard and Nananukul (2010) and by Adulyasak et al. (2014a). Bard and Nananukul (2010) introduced a branch-and-price procedure for the PRP with the ML policy and with multiple vehicles. Instances with up to 10 customers were solved to optimality within 30 minutes. Adulyasak et al. (2014a) considered both the multi-vehicle VMIRP and the MVPRP. They introduced different mathematical formulations, with and without a vehicle index, to solve the problems under both the ML and OU inventory replenishment policies. By using parallel computing, the algorithms of Adulyasak et al. (2014a) could solve instances with up to 45 and 50 customers (with three periods and three vehicles) for the VMIRP and PRP with the ML policy, respectively. For the OU policy, the algorithms could handle instances with up to 45 customers (with three periods and three vehicles) and 35 customers (with six periods and three vehicles) for the VMIRP and the PRP, respectively. Heuristic algorithms for both the multi-vehicle VMIRP and the MVPRP can be found in Adulyasak et al. (2014b).

According to the classification introduced by Coelho et al. (2014), all the inventory routing problems mentioned above belong to the class of problems with *finite* time horizon. The problems are also *deterministic* and *static* due to the assumption that consumption rates are known upfront. If the information on demand is not fully available to the decision maker at the beginning of the planning horizon, the problem is not deterministic. In this case, if the probability distribution of the demand is known, then the problem is called the stochastic IRP. Dynamic IRPs arise when demand is not fully known in advance, but is gradually revealed over time, as opposed to what happens in a static context. Solyalı et al. (2012) introduced a branch-and-cut algorithm for a robust IRP in which demand is uncertain but its probability distribution is unknown. This algorithm could solve instances with up to 30 customers and seven periods. Their approach was also adapted to solve the deterministic IRP with demand backlogging and could solve instances of the same size as for the robust IRP. For these last variants of the IRP, the reader is referred to the survey of Coelho et al. (2014).

1.2. Contributions

The main contribution of this paper is to propose a new exact algorithm for the MVPRP specifically tailored for the OU policy which is accounted to be the most difficult MVPRP policy (Adulyasak et al. 2014a). It is a Benders decomposition algorithm based on the path-based formulation proposed by Solyalı and Süral (2011) and also used by Adulyasak et al. (2014a). The problem is decomposed into a master problem, that decides whether to produce the product, the quantity to be produced, and the customers to be replenished for every period of the planning horizon and in a slave problem, that is further decomposed into Capacitated VRP, one problem for each period of the planning horizon. The algorithm relies on procedures used to compute a lower bound on the total routing cost of any optimal MVPRP solution and to generate a priori optimality cuts. In order to evaluate and assess the efficiency of our algorithm, extensive computational experiments were performed on both MVPRP and multi-vehicle VMIRP instances.

The remainder of this paper is organized as follows. Section 2 formally defines the problem and presents the formulation proposed by Adulyasak et al. (2014a). The Benders reformulation, the logic-based Benders decomposition algorithm, and its algorithmic features are then presented in Section 3. Section 4 introduces features that improve the efficiency of the algorithm. Section 5 presents the results of extensive computational experiments performed on both MVPRP and multi-vehicle VMIRP instances. Conclusions and future research directions are given in Section 6.

2. Problem description and mathematical formulation

In this section, we formally describe the MVPRP and we give the mathematical formulation of Adulyasak et al. (2014a) that is in turn based on the formulation of Solyalı and Süral (2011).

Let $G=(N,E)$ be a complete and undirected graph where $N=\{0,\dots,n\}$ is the node set and E is the edge set. Set $N_c=\{1,\dots,n\}$ corresponds to n *customers* and node 0 corresponds to the *plant* or *depot*. Each edge $\{i,j\}\in E$ is associated a *routing cost* c_{ij}. We consider the problem of supplying the customers with a single product from the plant over a discrete planning horizon of length l. At the beginning of each period, the production plant can produce at most C units of product with a fixed setup cost equal to f and a unit production cost equal to u. Node i, $i \in N$, has an inventory capacity of L_i units and an inventory unit cost equal to h_i. The quantity of the product held by node i at the beginning of the planning horizon is I_{i0}. Customer $i \in N_c$ requires d_{it} units of product during period t and can be visited at most once per period. A set of m identical vehicles of capacity Q are available at the depot.

The MVPRP consists of simultaneously deciding (i) when and how much to produce at the plant (ii) when and how much to deliver to each customer and (iii) what routes to use in every period of the planning horizon. The objective is to minimize the sum of the production, inventory and routing costs during the planning horizon without causing stockouts at any of the customers.

In this paper, we consider the OU policy as replenishment policy. Moreover, as commonly assumed in the literature, for each period the production at the plant takes place before delivery and the deliveries at the customers are executed at the beginning of the time period.

2.1. Mathematical formulation

The following notation is adopted, as defined by Adulyasak et al. (2014a):

- T: set of time periods $\{1,\ldots,l\}$; to simplify the formulation, we introduce a fictitious period $l+1$, and we define $T' = T \cup \{l+1\}$;
- g_{ivt}: total delivery quantity when customer i is visited in period t and the previous visit is in period v, computed as

$$g_{ivt} = \begin{cases} \sum_{j=1}^{t-1} d_{ij} + (L_i - I_{i0}), & if \quad v = 0, \\ \sum_{j=v}^{t-1} d_{ij}, & if \quad 0 < v < t \leq l+1; \end{cases}$$

- e_{ivt}: total inventory holding cost when customer i is visited in period t and the previous visit is in period v, computed as

$$e_{ivt} = \begin{cases} h_i \Big(\sum_{j=1}^{t-1} (I_{i0} - \sum_{r=1}^{j} d_{ir}) \Big), & if \quad v = 0, \\ h_i \Big(\sum_{j=v}^{t-1} (L_i - \sum_{r=v}^{j} d_{ir}) \Big), & if \quad 0 < v < t \leq l+1; \end{cases}$$

- $\mu(i,t)$: the latest period after period t when customer i can be replenished next without having a stockout, computed as $\mu(i,t) = \arg\max_{t<v\leq l+1}\{g_{itv} \leq L_i\}$;
- $\pi(i,t)$: the earliest period before period t when customer i can be replenished without having a stockout, computed as $\pi(i,t) = \arg\min_{0\leq v<t}\{g_{ivt} \leq L_i\}$.

Note that in the definitions listed above, $g_{iv\,l+1}$ is a fictitious delivery, and it is only used to compute $\mu(i,t)$ and $\pi(i,t)$. The following decision variables are used by the formulation:

- p_t: nonnegative continuous variable denoting the production quantity in period t;
- I_{it}: nonnegative continuous variable denoting inventory level at node i at the end of period t;

- y_t: binary variable equal to 1 if and only if there is production at the plant in period t;
- z_{it}: binary variable equal to 1 if and only if node i, $i \in N_c$, is visited in period t;
- s_t: nonnegative integer variable denoting the number of vehicles used in period t;
- x_{ijt}: integer variable that might take value in $\{0,1\}$, $\{i,j\} \in E$, $i \neq 0, t \in T$, and value in $\{0,1,2\}$, $\{0,j\} \in E, t \in T$;
- λ_{ivt}: binary variable equal to 1 if and only if node i is visited in period t and the previous visit is in period v.

The following additional notation is used in the formulation. We denote with $E(S)$ the set of edges with both nodes in S, i.e., $\{\{i,j\} \in E : i,j \in S, S \subseteq N\}$ and with $\delta(S)$ the set of edges incident to a node in S, i.e., $\delta(S) = \{\{i,j\} \in E : i \in S, j \notin S \textit{ or } i \notin S, j \in S\}$; we use $\delta(i)$ to denote the set of edges incident to node i. The mathematical formulation without a vehicle index of the MVPRP described by Adulyasak et al. (2014a) for the OU policy is as follows:

$$(F) \quad z(F) = \min \sum_{t \in T} \Big(up_t + fy_t + h_0 I_{0t} + \sum_{\{i,j\} \in E} c_{ij} x_{ijt} \Big) + \sum_{t \in T'} \sum_{i \in N_c} \sum_{v=\pi(i,t)}^{t-1} e_{ivt} \lambda_{ivt} \tag{1}$$

$$\text{s.t. } I_{0\,t-1} + p_t = \sum_{i \in N_c} \sum_{v=\pi(i,t)}^{t-1} g_{ivt} \lambda_{ivt} + I_{0t}, \quad \forall t \in T, \tag{2}$$

$$p_t \leq C y_t, \quad \forall t \in T, \tag{3}$$

$$I_{0t} \leq L_0, \quad \forall t \in T, \tag{4}$$

$$\sum_{v=\pi(i,t)}^{t-1} \lambda_{ivt} = z_{it}, \quad \forall i \in N_c, t \in T, \tag{5}$$

$$\sum_{t=1}^{\mu(i,0)} \lambda_{i0t} = 1, \quad \forall i \in N_c, \tag{6}$$

$$\sum_{v=\pi(i,t)}^{t-1} \lambda_{ivt} - \sum_{v=t+1}^{\mu(i,t)} \lambda_{itv} = 0, \quad \forall i \in N_c, t \in T, \tag{7}$$

$$\sum_{t=\pi(i,l+1)}^{l} \lambda_{it\,l+1} = 1, \quad \forall i \in N_c, \tag{8}$$

$$\sum_{\{j,j'\} \in \delta(0)} x_{jj't} = 2 s_t, \quad t \in T, \tag{9}$$

$$\sum_{\{j,j'\} \in \delta(i)} x_{jj't} = 2 z_{it}, \quad \forall i \in N_c, t \in T, \tag{10}$$

$$s_t \leq m, \quad \forall t \in T, \tag{11}$$

$$Q \sum_{\{i,j\} \in E(S)} x_{ijt} \leq \sum_{i \in S} \Big(Q z_{it} - \sum_{v=\pi(i,t)}^{t-1} g_{ivt} \lambda_{ivt} \Big), \; \forall S \subseteq N_c, |S| \geq 2, t \in T, \tag{12}$$

$$p_t, I_{it} \geq 0, \quad \forall i \in N, t \in T, \tag{13}$$

$$y_t, z_{it} \in \{0,1\}, \quad \forall i \in N_c, t \in T, \tag{14}$$

$$\lambda_{ivt} \in \{0,1\}, \quad \forall i \in N_c, v, t \in T', \tag{15}$$

$$s_t \in \{0,\dots,m\}, \quad \forall t \in T, \tag{16}$$

$$x_{ijt} \in \{0,1\}, \quad \forall \{i,j\} \in E : i \neq 0, t \in T, \tag{17}$$

$$x_{0jt} \in \{0,1,2\}, \quad \forall j \in N_c, t \in T. \tag{18}$$

The objective function (1) minimizes the total cost consisting of production, setup, inventory, and routing costs. Constraints (2) guarantee the inventory flow balance at the plant. Constraints (3) ensure that the production capacity is not violated if the production takes place, otherwise no product is produced. The inventory capacity constraint at the plant for each period is imposed by constraints (4). Constraints (5) link variables z with variables λ. Constraints (6), (7) and (8) are flow conservation constraints on the shortest path network for each customer. Constraints (9), (10) and (12) are the routing related constraints. Constraints (11) limit the number of available vehicles. Constraints (12) impose both connectivity and vehicle capacity constraints and are based on the the generalized fractional subtour elimination constraints of the Capacitated VRP (Toth and Vigo 2014). Given a feasible solution F and a pair (S,t), the term at the right-hand side is equal to Q multiplied by the number of customers in S visited on day t (say α) minus the total demand delivered to the visited customers. If the inequality is divided by Q, the term at the left-hand side, i.e., the total number of edges in solution with both nodes in set S, must be less than or equal to α minus a lower bound on the number of vehicles necessary to visit the customers in S (say β, computed as the ratio between the total demand in S and Q), thus imposing that at least $\lceil \beta \rceil$ edges leave the customer set S.

3. Benders decomposition

In this section, we describe an exact approach based on logic-based Benders decomposition, that was formally developed by Hooker (2000), and applied with success by Hooker and Ottosson (2003) to 0-1 programming and by Hooker (2007) to planning and scheduling problems. The approach was later specialized to mixed integer programming by Codato and Fischetti (2006) who introduced the so-called combinatorial Benders cuts. Logic-based Benders decomposition is a generalization of classical Benders decomposition that can be applied to a much wider variety of combinatorial optimization problems since the subproblem may be any combinatorial problem, not necessarily a linear or nonlinear programming problem (Hooker and Ottosson 2003).

3.1. Logic-based Benders reformulation

Benders decomposition was first proposed by Benders (1962) to efficiently solve mixed integer programming models. Basically, it decomposes the original problem into two simpler ones, i.e. an integer master problem (BMP) and a linear slave problem or subproblem (BSP), which are solved in an iterative fashion by utilizing the solution of one in the other. At each iteration, the master problem actually behaves as a relaxation of the original problem and provides fixed integer variable values for the slave problem to obtain a feasible solution. A Benders cut is constructed and added to the master problem in the next iteration to exclude the solution just obtained in the last master problem. Therefore, each solution of the master problem must satisfy all the Benders cuts generated so far to avoid repetition. The master problem and the slave problem are solved in this iterative fashion until an optimal solution to the original problem is obtained. The generation of Benders cuts (i.e., *optimality* and *feasibility cuts*) is the core of the Benders decomposition algorithm, and they guarantee the convergence of the iterations to the optimal solution of the original problem. Furthermore, these cuts also determine how fast the algorithm converges. The classic Benders decomposition algorithm was proposed for linear programming problems (Benders 1962), the cut generation of which is based on the strong duality property of linear programming. Geoffrion (1972) has extended it to a larger class of mathematical programming problems. For more general integer programming, logic-based Benders decomposition was proposed to generate valid integer Benders cuts (Hooker 2000, Hooker and Ottosson 2003). The key is to generalize the linear programming dual used in the classical method to an inference dual and the solution of the inference dual takes the form of a logical deduction that yields valid Benders cuts. In the following, we describe in details our reformulation of formulation F based on logic-based Benders decomposition.

The reformulation is based on the observation that once variables p, I, y, λ and z have been fixed, formulation F decomposes into l subproblems, where each subproblem is a CVRP defined on the customer delivery quantities determined by the values of variables z and λ.

Introducing nonnegative continuous extra variables ω_t, $\forall t \in T$, for the routing costs associated with the different periods, we can reformulate the MVPRP as follows:

$$(F) \quad z(F) = \min \sum_{t\in T}\Big(up_t + fy_t + h_0 I_{0t} + \omega_t\Big) + \sum_{t\in T'}\sum_{i\in N_c}\sum_{v=\pi(i,t)}^{t-1} e_{ivt}\lambda_{ivt} \tag{19}$$

$$\text{s.t.} \sum_{t\in T}\omega_t \geq \phi(z,\lambda),$$

$$(2)-(8), (13), (14), (15),$$

$$\omega_t \geq 0, \forall t \in T, \tag{20}$$

where

$$\phi(z,\lambda) = \min \sum_{t\in T}\sum_{\{i,j\}\in E} c_{ij}x_{ijt} \tag{21}$$

$$\text{s.t. } (9)-(12), (16), (17) \text{ and } (18),$$

is the *separation subproblem*, where we assume $\phi(z,\lambda) = \infty$ if the problem is infeasible.

The above reformulation can be handled by solving a Benders master problem (BMP) to integer optimality before calling a Benders subproblem (BSP) corresponding to the separation subproblem. We have three possible outcomes about a solution $(p^*, I^*, y^*, z^*, \lambda^*, \omega^*)$ of BMP:

i) Problem BSP is infeasible for (z^*, λ^*), and one can add the following *infeasibility* cut to BMP

$$f_{\lambda^*}(\lambda) \geq 1 \tag{22}$$

where

$$f_{\lambda^*}(\lambda) = \sum_{t\in T}\sum_{i\in N_c}\sum_{v=\pi(i,t):\lambda^*_{ivt}=1}^{t-1}(1-\lambda_{ivt}) + \sum_{t\in T}\sum_{i\in N_c}\sum_{v=\pi(i,t):\lambda^*_{ivt}=0}^{t-1}\lambda_{ivt}$$

that cuts off solution λ^*.

ii) Problem BSP is feasible for (z^*, λ^*), but $\phi(z^*, \lambda^*) > \sum_{t\in T}\omega^*_t$, then one can add the following *optimality* cut to BMP

$$\sum_{t\in T}\omega_t \geq g_{(z^*,\lambda^*)}(\lambda) \tag{23}$$

where $g_{(z^*,\lambda^*)}(\lambda) = \phi(z^*, \lambda^*) - (\phi(z^*, \lambda^*) - LB_R)f_{\lambda^*}(\lambda)$ and LB_R is a lower bound on $\phi(z, \lambda)$, i.e., a lower bound on the routing cost of any optimal MVPRP solution.

iii) Problem BSP is feasible for (z^*, λ^*), and $\phi(z^*, \lambda^*) = \sum_{t\in T}\omega^*_t = \sum_{t\in T}\sum_{\{i,j\}\in E} c_{ij}x^*_{ijt}$, where (s^*, x^*) is an optimal solution of BSP. Solution $(p^*, I^*, y^*, z^*, \lambda^*, s^*, x^*)$ is an optimal MVPRP solution with value $\sum_{t\in T}\left(up^*_t + fy^*_t + h_0 I^*_{0t}\right) + \sum_{t\in T'}\sum_{i\in N_c}\sum_{v=\pi(i,t)}^{t-1} e_{ivt}\lambda^*_{ivt} + \phi(z^*, \lambda^*)$.

The exact algorithm based on the above logic-based Benders reformulation is as follows. The initial master problem BMP is defined by the objective function (19) subject to constraints (2)-(8), (13), (14), (15) and (20). The algorithm performs the following steps:

1. *Initialization.* Compute lower bound LB_R (see Section 3.3) and set $LB = 0$ and $UB = \infty$.
2. While $LB < UB$
   a. *Solution of the master problem.* Solve problem BMP. If BMP is infeasible, no feasible MVPRP solution exists, stop. Otherwise, let $(p^*, I^*, y^*, z^*, \lambda^*, \omega^*)$ be the corresponding optimal solution of cost $\overline{z}$. Set $LB = \overline{z}$.
   b. *Solution of the subproblem.* Solve problem BSP; there are two possible outcomes:
      i. BSP is infeasible. Add the infeasibility cut (22) to BMP;
      ii. BSP admits a feasible and integer solution such that $\phi(z^*, \lambda^*) > \sum_{t\in T}\omega^*_t$. Add the optimality cut (23) to BMP.

      Let $(p^*, I^*, y^*, z^*, \lambda^*, s^*, x^*)$ be the corresponding MVPRP feasible solution of cost $\hat{z} = \sum_{t\in T}\left(up^*_t + fy^*_t + h_0 I^*_{0t}\right) + \sum_{t\in T'}\sum_{i\in N_c}\sum_{v=\pi(i,t)}^{t-1} e_{ivt}\lambda^*_{ivt} + \phi(z^*, \lambda^*)$. Set $UB = \min\{UB, \hat{z}\}$.

At the different iterations of the above algorithm, LB represents a valid lower bound on $z(F)$ whereas at the end of the algorithm $UB = z(F)$ (we assume $z(F) = \infty$ if problem F does not admit a feasible solution). In Section 3.2, we describe in details how problem BSP is solved at Step 2.b and how Step 2.b.i is implemented in practice.

The following lemma holds about optimality cut (23).

Lemma 1 *The optimality cut* (23) *satisfies the following two properties:*

(i) $g_{(z^*,\lambda^*)}(\lambda^*) = \phi(z^*,\lambda^*)$*;*

(ii) Any feasible solution $(\overline{p},\overline{I},\overline{y},\overline{z},\overline{\lambda},\overline{\omega},\overline{s},\overline{x})$ *of problem* F *such that* $\overline{\lambda} \neq \lambda^*$ *satisfies* $\sum_{t\in T}\sum_{\{i,j\}\in E} c_{ij}\overline{x}_{ijt} \geq g_{(z^*,\lambda^*)}(\overline{\lambda})$.

Proof. Property (i) follows directly from the definition of the optimality cut. Regarding property (ii), since $\overline{\lambda} \neq \lambda^*$ we have $f_{\lambda^*}(\overline{\lambda}) \geq 1$, and

$$\phi(z^*,\lambda^*) - (\phi(z^*,\lambda^*) - LB_R) f_{\lambda^*}(\overline{\lambda}) \leq \phi(z^*,\lambda^*) - (\phi(z^*,\lambda^*) - LB_R) \leq LB_R,$$

and the inequality $\sum_{t\in T}\sum_{\{i,j\}\in E} c_{ij}\overline{x}_{ijt} \geq g_{(z^*,\lambda^*)}(\overline{z})$ holds since we have $\sum_{t\in T}\sum_{\{i,j\}\in E} c_{ij}\overline{x}_{ijt} \geq LB_R$ due to the definition of LB_R. □

The following theorem then shows the correctness of the algorithm.

Theorem 1 *The logic-based Benders algorithm terminates after finitely many steps.*

Proof. Suppose first that the algorithm terminates with a finite solution UB. Clearly, UB is an upper bound on the solution cost $z(F)$. Because the algorithm terminated, we have $LB = UB$ and due to Lemma 1, property (ii), LB is a valid lower bound on the optimal solution cost. Because the solution corresponding to UB is feasible, it is also optimal. Secondly, since the domain of BMP variables z and λ is finite, only finitely many subproblems can be defined (and corresponding Benders infeasibility and optimality cuts), and the optimal value is reached after finitely many steps. If no feasible MVPRP solution exists, then the algorithm will terminate at Step 2.a after finitely many steps; this is due, again, to the fact that the domain of BMP variables z and λ is finite. □

It is worth mentioning that the decomposition approach used to solve the MVPRP strongly relies on the structural properties of the OU policy, i.e., on the definition of variables λ_{ivt} that enables us to formulate and solve the corresponding subproblem in an effective and efficient way. More precisely, under the OU policy, the set $\mathcal{Q}_i$ of all possible demand values for a customer $i \in N_c$ is polynomially sized, being $\mathcal{Q}_i$ equal to $\{g_{ivt} \leq Q : \forall v \in \{0\} \cup T, \forall t \in T', t > v\}$, a property that is no more valid under the ML policy, being in this case sets $\mathcal{Q}_i$ pseudo-polynomially sized since general nonnegative decision variables representing the quantities delivered to the customers over the planning horizon are necessary to model the problem. In the case of the ML policy, if we are willing to sacrifice the property of having only a polynomial number of variables in the master problem by using variables λ_{iwt} instead of variables λ_{ivt}, where λ_{iwt} is equal to 1 if a quantity w is delivered to customer i in period t, and variables $q_{it} = \sum_{w=0}^{Q} w\lambda_{iwt}$ are used to denote the quantity delivered in period t to customer i, then variables λ_{iwt} can still be used similarly as variables λ_{ivt} to model the subproblem, and derive the corresponding infeasibility and optimality cuts. Nevertheless, the complexity of solving the master is greatly increased compared to the OU policy.

3.2. Solving the subproblem

In this section, we describe the procedure used at Step 2.b of the exact algorithm to solve problem BSP defined by the objective function (21) subject to constraints (9)-(12), (16), (17) and (18).

Given a BMP solution $(\overline{p}, \overline{I}, \overline{y}, \overline{z}, \overline{\lambda}, \overline{\omega})$, problem BSP decomposes into l subproblems, where each subproblem corresponds to a CVRP. More precisely, the CVRP associated with period $t \in T$, denoted CVRP(t), is defined on a complete and undirected graph $G^t = (V^t, E^t)$ where $V^t = \{0\} \cup N_c^t$ is the vertex set and E^t is the edge set defined as $E^t = \{\{i,j\} : i,j \in V^t, i < j\}$. Set N_c^t is defined as $\{i \in N_c : \overline{z}_{it} = 1\}$, i.e., is the set of customers serviced in period t, whereas vertex 0 corresponds to the depot of graph G. A nonnegative *cost*, $c_{ij}^t = c_{ij}$, is associated with each edge $\{i,j\}$. Each customer $i \in N_c^t$ is associated with a known nonnegative *demand*, $q_{it} = g_{i\overline{v}_i t}$, to be delivered where $\overline{v}_i$ is such that $\overline{\lambda}_{i\overline{v}_i t} = 1$. A set of m identical vehicles, each with capacity Q, is available at the depot. The problem consists of finding a collection of at most m simple *cycles* or *routes* with minimum cost, defined as the sum of the costs of the edges belonging to the routes, and such that: (i) each route visits the depot vertex, (ii) each customer vertex is visited by exactly one route, and (iii) the sum of the demands of the vertices visited by a route does not exceed the vehicle capacity Q.

Solving problem CVRP(t) To solve each problem CVRP(t) we use an exact algorithm based on the set partitioning formulation. The reader is referred to Poggi and Uchoa (2014), Zhang et al. (2019) for a review of exact methods based on the set partitioning formulation.

For sake of simplicity, we omit the index t in description of the formulation. Let $\mathcal{R}$ be the index set of all feasible routes and let a_{ir} be a binary coefficient that is equal to 1 if vertex $i \in N_c^t$ belongs to route $r \in \mathcal{R}$ and takes the value 0 otherwise (note that $a_{0r} = 1, \forall r \in \mathcal{R}$). Each route $r \in \mathcal{R}$ has an associated cost b_r, that is equal to the optimal solution cost of the TSP instance defined by route r. Let ξ_r be a binary variable that is equal to 1 if and only if route $r \in \mathcal{R}$ belongs to the optimal solution. The formulation for the CVRP(t) is as follows:

$$CVRP(t) \quad \omega_t = \min \quad \sum_{r \in \mathcal{R}} b_r \xi_r$$

$$\text{s.t.} \sum_{r\in\mathcal{R}} a_{ir}\xi_r = \overline{\lambda}_{i\overline{v}_i t}, \quad \forall i \in N_c^t, \tag{24}$$

$$\sum_{r\in\mathcal{R}} \xi_r \le m, \tag{25}$$

$$\xi_r \in \{0,1\}, \quad \forall r \in \mathcal{R}.$$

Constraints (24) specify that each customer $i \in N_c^t$ must be covered by one route and constraint (25) requires that at most m routes are selected.

Problem CVRP(t) can be infeasible due to the definition of set N_c^t and of the limited number of vehicles m and, therefore, a corresponding infeasibility cut must be added to BMP. To check if CVRP(t) admits a feasible solution, we solve the Bin Packing Problem (BPP) instance defined by $|N_c^t|$ items, weights $\{q_{it}\}$ and with at most m bins of capacity equal to Q. If the resulting BPP instance is infeasible, then the following infeasibility cut can be defined:

$$\sum_{i\in N_c:\overline{z}_{it}=1} (1-\lambda_{i\overline{v}_i t}) + \sum_{i\in N_c:\overline{z}_{it}=1} \sum_{v=\pi(i,t):\overline{\lambda}_{ivt}=0}^{t-1} \lambda_{ivt} \ge 1,$$

that cuts off solution $\{\overline{\lambda}_{ivt}\}$.

The above cut can be strengthened by observing that if solution $\{\overline{\lambda}_{ivt}\}$ is infeasible, then also a solution $\{\lambda_{ivt}\}$ is infeasible whenever $\lambda_{ivt} \ge \overline{\lambda}_{ivt}$, $\forall i \in N_c$, $v = \pi(i,t), \ldots, t-1$, i.e., deliver to additional customers on the same day will also result in an infeasible solution. Hence, the following infeasibility cut can be added to BMP:

$$\sum_{i\in N_c:\overline{z}_{it}=1} (1-\lambda_{i\overline{v}_i t}) \ge 1. \tag{26}$$

The above cut can be lifted by observing that for a given $i \in N_c$ we have $g_{iv_1t} > g_{iv_2t}$ if $v_1 < v_2 < t$, i.e., if the previous visit to customer i is in between $[\pi(i,t), \overline{v}_i - 1]$, then the demand to be delivered to i at period t will be greater than the demand delivered if the previous visit is at period $\overline{v}_i$. Hence, the following strengthened infeasibility cut can be derived:

$$\sum_{i\in N_c:\overline{z}_{it}=1} \sum_{v=\pi(i,t)}^{\overline{v}_i} (1-\lambda_{ivt}) \ge 1. \tag{27}$$

If the BPP instance defined above admits a feasible solution, then problem CVRP(t) admits a finite optimal solution. In the rest of this section, we briefly describe the method used to solve to optimality problem CVRP(t). The method is based on the *route enumeration* procedure described by Baldacci et al. (2008) and on the hybrid strategy used in Pessoa et al. (2009) and Pessoa et al. (2008). Since solving problem CVRP(t) can be time consuming, we also consider the generation of valid optimality cuts based on the LP-relaxation of formulation CVRP(t).

Let $u = (u_0, u_{i_1}, \ldots, u_{i_{|N_c^t|}})$ be a vector of dual variables, where u_{i_h}, $i_h \in N_c^t$, and $u_0 \leq 0$ are associated with constraints (24) and (25), respectively. The dual of the LP-relaxation of problem CVRP(t) is as follows:

$$\begin{aligned} DCVRP(t) \quad \max \quad & \sum_{i \in N_c^t} \overline{\lambda}_{i\overline{v}_i t} u_i + m u_0 \\ \text{s.t.} \quad & \sum_{i \in N_c^t} a_{ir} u_i + u_0 \leq b_r, \quad \forall r \in \mathcal{R}, \\ & u_i \in \mathbb{R}, \quad \forall i \in N_c^t, \\ & u_0 \leq 0. \end{aligned}$$

The exact algorithm used to solve problem CVRP(t) to optimality is as follows.

1. *Compute a primal bound.* Compute a primal bound z_{UB} on the optimal solution cost ω_t by means of a tabu search heuristic based on the algorithm proposed by Gendreau et al. (1994).
2. *Solve the LP-relaxation of problem CVRP(t).* Solve the LP-relaxation of problem CVRP(t) by means of standard column generation procedure. Let z_{LP} be the optimal solution cost and ξ^* and u^* be the corresponding primal and dual solutions, respectively. The restricted master problem is initialized with the set of routes forming the primal solution computed at Step 1 and procedure GENROUTE described in Baldacci et al. (2008) is used to generate feasible CVRP routes.
3. *Add an optimality cut.* If $z_{LP} > \overline{\omega}_t$, the following optimality cut is added to BMP:

$$\omega_t \geq \sum_{i \in N_c^t} u_i^* \lambda_{i\overline{v}_i t} + m u_0^*, \tag{28}$$

   and the procedure terminates, otherwise the next step is executed.
4. *Route enumeration.* Let $\overline{b}_r = b_r - \sum_{i \in N_c^t} a_{ir} u_i^* - u_0^*$ be the reduced cost associated with route $r \in \mathcal{R}$ with respect to the dual solution u^*. Using procedure GENROUTE, generate the largest subset $\overline{\mathcal{R}}$ of the route set $\mathcal{R}$ such that:

$$\left.\begin{aligned} & |\overline{\mathcal{R}}| \leq \Delta, \\ & \max_{r \in \overline{\mathcal{R}}} \{\overline{b}_r\} \leq z_{UB} - z_{LP}. \end{aligned}\right\}$$

   where Δ is a user-defined parameter, that is set equal to 60000 in the computational experiments reported in Section (6).
5. *Solve problem CVRP(t).* We have the following two cases:
   i) If $|\overline{\mathcal{R}}| < \Delta$, solve the reduced problem obtained from problem CVRP(t) by substituting the route set $\mathcal{R}$ with set $\overline{\mathcal{R}}$ by the generic branch-and-cut algorithm of the IBM Cplex solver (IBM CPLEX 2016).

ii) If $|\overline{\mathcal{R}}| = \Delta$. Solve problem CVRP($t$) using a branch-and-price algorithm where procedure GENROUTE is again used to generate feasible routes in a column generation fashion and where the branching on sets strategy is used (Lysgaard et al. 2004).

Optimally cut (28), can be lifted by using the same observation used to lift infeasibility cut (27) as follows:

$$\omega_t \geq \sum_{i \in N_c^t} \sum_{v=\pi(i,t)}^{\overline{v}_i} u_i^* \lambda_{ivt} + m u_0^*. \tag{29}$$

Updating problem BMP. Problems CVRP(t), $\forall t \in T$, are first checked for feasibility and any violated infeasibility cut (27) is added to BMP. If at least one infeasibility cut has been detected, the procedure terminates and the master problem BMP is solved again at the next main iteration. Otherwise, if all problems CVRP(t), $\forall t \in T$, are feasible, then the problems are solved in sequence (for $t = 1, \ldots, l$) up to Step 3 of the exact algorithm used to solve problems CVRP(t) and any violated optimality cut (29) is added to BMP. Also in this case, if any violated cut is found, the procedure terminates and the master problem BMP is solved again at the next main iteration.

If no infeasibility (27) and optimality (29) cuts have been detected, then all problems CVRP(t) are solved to optimality by executing steps 4 and 5 of the exact algorithm, and Step 2.b.ii of the exact algorithm for the MVPRP is then executed to check if a new optimality cut (23) must be added to BMP.

3.3. Computing lower bound LB_R on the routing cost

In this section, we describe the relaxation and the bounding procedure used to compute lower bound LB_R introduced to define the optimality cut (23).

Let f_{it}, $i \in N_c$, $t \in T$, be a lower bound on the number of visits that customer i must receive up to period t. For each $i \in N_c$ and $t \in T$, let $\overline{q}_{it}$ be the cumulative demand of customer i up to period t computed as $\overline{q}_{it} = \max\{0, -I_{i0} + \sum_{v=1}^{t} d_{iv}\}$ for $t = 1, \ldots, l$. Values f_{it}, $\forall i \in N_c$, $t \in T$, can be computed as $f_{it} = \lceil \overline{q}_{it} / \min\{Q, L_i\} \rceil$. In addition, let m_L and m_U be lower and upper bounds on the total number of vehicles needed in the planning horizon, respectively. Let $\hat{q}_i = \sum_{t \in T} d_{it} - I_{i0}$ be the total quantity of product required by customer $i \in N_c$ over the planning horizon. Values m_L and m_U can be defined as $m_L = \left\lceil \frac{1}{Q} \sum_{i \in N_c} \hat{q}_i \right\rceil$ and $m_U = m \cdot l$. The relaxation also requires the definition of value q_i, $\forall i \in N_c$, defined as a lower bound on the quantity delivered to customer i during any visit in the planning horizon that can be computed as $q_i = \min_{t \in T}\{d_{it}\}$, $i \in N_c$.

We define a route as a least cost simple cycle of graph G passing through depot 0 and such that the total demand of the customers visited computed using demands $\{q_i\}$ does not exceed the vehicle capacity Q. Let $\tilde{\mathcal{R}}$ be the index set of all routes and let a_{ir} be a binary coefficient that is equal to 1 if vertex $i \in N_c$ belongs to route $r \in \tilde{\mathcal{R}}$ and takes the value 0 otherwise. Each route $r \in \tilde{\mathcal{R}}$ has an associated routing cost b_r.

A valid lower bound on the routing cost of any optimal MVPRP solution is given by the optimal solution cost of the following integer problem:

$$(RF) \quad z(RF) = \min \sum_{r\in\tilde{\mathcal{R}}} b_r \xi_r$$

$$\text{s.t.} \sum_{r\in\tilde{\mathcal{R}}} a_{ir}\xi_r \geq f_{il} \quad \forall i \in N_c, \tag{30}$$

$$\sum_{r\in\tilde{\mathcal{R}}} \xi_r \geq m_L, \tag{31}$$

$$\sum_{r\in\tilde{\mathcal{R}}} \xi_r \leq m_U, \tag{32}$$

$$\xi_r \geq 0 \text{ integer }, \forall r \in \tilde{\mathcal{R}}.$$

Lower bound LB_R is computed by a bounding procedure as a near-optimal dual solution of the LP-relaxation of problem RF. The procedure differs from standard column generation methods based on the simplex algorithm as it uses a dual ascent heuristic to solve the master problem (see Baldacci et al. 2010). The bounding procedure is based on the following theorem.

Theorem 2 *Let $u_i \geq 0$, $i \in N_c$, $v_L \geq 0$, and $v_U \leq 0$ be the dual variables associated with constraints* (30)*,* (31) *and* (32)*, respectively. Associate penalties $\lambda_i \geq 0$, $i \in N_c$, with constraints* (30) *and $w_L \geq 0$, and $w_U \leq 0$ with constraints* (31) *and* (32)*, respectively. Let ϕ_i, $i \in N_c$, be computed as $\phi_i = q_i \min_{r\in\tilde{\mathcal{R}}_i}\{(b_r - \lambda(r) - w_L - w_U)/q(r)\}$, where $\tilde{\mathcal{R}}_i \subseteq \tilde{\mathcal{R}}$ is the index set of routes passing through customer $i \in N_c$, $\lambda(r) = \sum_{i\in N_c} a_{ir}\lambda_i$ and $q(r) = \sum_{i\in N_c} a_{ir}q_i$. A feasible dual solution (u, v_L, u_U), of cost $z(\lambda, w_L, w_U)$ can be computed by means of the following expressions:*

$$u_i = \phi_i + \lambda_i, i \in N_c, v_L = w_L, \mathit{and}\ v_U = w_U. \tag{33}$$

Proof. Consider the dual constraint corresponding to route $r \in \tilde{\mathcal{R}}$. Since for each i visited by route r we have $r \in \tilde{\mathcal{R}}_i$, we have

$$\phi_i = q_i \min_{r'\in\tilde{\mathcal{R}}_i}\{(b'_r - \lambda(r') - w_L - w_U)/q(r')\} \leq q_i(b_r - \lambda(r) - w_L - w_U)/q(r).$$

Hence

$$\sum_{i\in N_c} a_{ir}u_i + v_L + v_U \leq \sum_{i\in N_c} a_{ir}q_i(b_r - \lambda(r) - w_L - w_U)/q(r) + \lambda(r) + w_L + w_U =$$
$$q(r)(b_r - \lambda(r) - w_L - w_U)/q(r) + \lambda(r) + w_L + w_U = b_r. \square$$

The optimal solution cost of the following problem

$$\max_{\lambda, w_L, w_U}\{z(\lambda, w_L, w_U)\} \tag{34}$$

provides the best possible lower bound which can be computed by means of Theorem 2. Problem (34) cannot be solved directly as the computation of solution u, for given penalties λ, w_L and w_U, requires the a priori generation of the set $\tilde{\mathcal{R}}$. In practice, we use an iterative algorithm which computes a lower bound as the cost of a suboptimal solution of problem (34) by using a limited subset of set $\tilde{\mathcal{R}}$ and by changing the values of vector λ, w_L and w_U. At each iteration, the procedure uses expressions (33) to find a solution (u, v_L, v_U) of the reduced problem defined on a route subset of $\tilde{\mathcal{R}}$. In addition, subgradient vectors are computed and used to change vector λ and w_L and w_U to maximize the value of the lower bound. In the procedure, we further relax the requirement that a route is a simple cycle of graph G and we extend the route set $\tilde{\mathcal{R}}$ to contain ng-routes (see Baldacci et al. 2011). This relaxation allows us to execute in pseudo polynomial time the pricing algorithm used to identify the route subset whose dual constraints are violated by the current solution (u, v_L, v_U). The above procedure is executed for a fixed number of iterations and lower bound LB_R is set equal to the maximum of the lower bounds computed at the different iterations.

4. Improving the Benders master problem

In this section, we describe valid inequalities to strengthen the Benders master problem. In particular, Section 4.1 describes two optimality cuts for the master problem used to accelerate the convergence of the exact algorithm.

Adulyasak et al. (2014a) described a number of valid inequalities to strengthen the LP-relaxation of formulation F. Among the different valid inequalities described by Adulyasak et al., we added to the Benders master problem the following inequality:

$$\sum_{t=1}^{t'} y_t \geq 1, \tag{35}$$

where t' is the earliest period when the plant must produce to prevent a stockout, computed as

$$t' = \arg\min_{1 \leq t \leq l} \left\{ \sum_{i \in N_c} \max\left\{0, \sum_{v=1}^{t} d_{iv} - I_{i0}\right\} - I_{00} > 0 \right\}.$$

We also added to the master problem the following valid inequalities

$$\sum_{i \in N_c} \sum_{v=\pi(i,t)}^{t-1} g_{ivt}\lambda_{ivt} \leq mQ, \quad \forall t \in T, \tag{36}$$

and

$$\sum_{i \in N_c} \sum_{v=\pi(i,t): g_{ivt} > \lceil Q/2 \rceil}^{t-1} \lambda_{ivt} \leq m, \quad \forall t \in T, \tag{37}$$

both based on the fact that a limited number of m vehicles are available in each period to serve the customers.

It is worth mentioning that, due to constraints (5), imposing integrality on variables λ ensures the integrality of variables z. Solyalı and Süral (2011) observed that also the vice versa holds. We conducted preliminary experiments to define the integrality requirements of variables z and λ, and as a result of our experiments, in the computational experiments reported in Section 6 we decided to impose integrality requirements on variables λ only.

4.1. Initial set of optimality cuts

In this section, we describe two ways of generating initial optimality cuts for the Benders master problem. The cuts are based on the following alternative formulation of problem CVRP(t).

Let $\hat{\mathcal{R}}$ be the index set of all feasible routes for period t where the demand q_{it} associated with customer i, $i \in N_c$, belongs to the discrete set of demands $\{g_{ivt} : v = \pi(i,t), \ldots, t-1\}$, i.e., the set of all possible demands of customer i visited in period t according to the set $\{\pi(i,t), \ldots, t-1\}$ of its potential previous visiting periods. Let a_{ivr} be a binary coefficient that is equal to 1 if and only if customer i belongs to route r and has associated a demand equal to g_{ivt}. Problem CVRP(t) can be formulated as follows:

$$
\begin{aligned}
CVRP(t) \quad \omega_t = \min \quad & \sum_{r \in \hat{\mathcal{R}}} b_r \xi_r \\
\text{s.t.} \quad & \sum_{r \in \hat{\mathcal{R}}} a_{ivr} \xi_r = \overline{\lambda}_{ivt}, \quad \forall i \in N_c,\ v = \pi(i,t), \ldots, t-1, && (38) \\
& \sum_{r \in \hat{\mathcal{R}}} \xi_r \leq m, && (39) \\
& \xi_r \in \{0,1\}, \quad \forall r \in \hat{\mathcal{R}},
\end{aligned}
$$

where constraints (38) state that a route servicing customer i with demand g_{ivt} must be in solution in period t if and only if $\overline{\lambda}_{ivt}$ is equal to 1, i.e., the previous visit of customer i occurred in period $v < t$. The procedure used to derive the optimality cuts is based on the following proposition.

Proposition 1 *Let problem P be defined as (P) $\min\{cx : Ax = b, x \in \mathbb{R}^n_+\}$, with $A \in \mathbb{R}^{m \times n}$, $c \in \mathbb{R}^n$, $b \in \mathbb{R}^m$, and let D be its dual, i.e., (D) $\max\{wb : wA \leq c, w \in \mathbb{R}^m\}$, where w is the dual vector associated with constraints $Ax = b$. Then any feasible dual solution of the following problem, $(P')\min\{cx : a^i x = \overline{b}_i, \forall i \in I_1, a^i x \geq \overline{b}_i, \forall i \in I_2, x \in \mathbb{R}^n_+\}$, where I_1 and I_2 form a partition of the index set $\{1, \ldots, m\}$ of the rows of matrix A, a^i denotes row i of matrix A, $\overline{b} \in \mathbb{R}^m$, is also a feasible solution of problem D.*

Proof. Let u be a feasible solution of the dual of problem P', that is $(D')\max\{u\overline{b} : uA \leq c, u_i \in \mathbb{R}, \forall i \in I_1, u_i \geq 0, \forall i \in I_2\}$. It is easy to see that the solution w of D obtained by setting $w_i = u_i$, $i = 1, \ldots, n$, is a feasible D solution. □

Let $w_{iv} \in \mathbb{R}$, $\forall i \in N_c$, $v = \pi(i,t),\ldots,t-1$, and $u_0 \leq 0$ be the dual variables associated with constraints (38) and (39), respectively. The dual of the LP-relaxation of CVRP(t) is as follows:

$$
\begin{aligned}
DCVRP(t) \quad \max \quad & \sum_{i\in N_c}\sum_{v=\pi(i,t)}^{t-1} \overline{\lambda}_{ivt} w_{iv} + m u_0 \\
\text{s.t.} \quad & \sum_{i\in N_c}\sum_{v=\pi(i,t)}^{t-1} a_{ivr} w_{iv} + u_0 \leq b_r, \quad \forall r \in \hat{\mathcal{R}}, \\
& w_{iv} \in \mathbb{R}, \quad \forall i \in N_c,\ v = \pi(i,t),\ldots,t-1, \\
& u_0 \leq 0.
\end{aligned}
$$

Type I cut Type I cut is based on the following LP problem derived from problem CVRP(t):

$$
\begin{aligned}
(T1) \quad \min \quad & \sum_{r\in\hat{\mathcal{R}}} b_r \xi_r \\
\text{s.t.} \quad & \sum_{r\in\hat{\mathcal{R}}} a_{ivr}\xi_r \geq 1, \quad \forall i \in N_c,\ v = \pi(i,t),\ldots,t-1, && (40) \\
& \sum_{r\in\hat{\mathcal{R}}} \xi_r \leq m, && (41) \\
& \xi_r \geq 0, \quad \forall r \in \hat{\mathcal{R}}.
\end{aligned}
$$

Let $w_{iv} \geq 0$, $\forall i \in N_c$, $v = \pi(i,t),\ldots,t-1$, and $u_0 \leq 0$ be the dual variable associated with constraints (40) and (41), respectively, and let (w^*, u_0^*) be a feasible dual solution of problem $T1$. Based on Proposition 1, the following inequality provides a valid optimality cut:

$$
\omega_t \geq \sum_{i\in N_c}\sum_{v=\pi(i,t)}^{t-1} w_{iv}^* \lambda_{ivt} + m u_0^*. \tag{42}
$$

Type II cut Type II cut is based on the LP problem derived from formulation F as follows.

Let $\hat{\mathcal{R}}_t$ be the index set of all feasible routes for period t where the demand q_{it} associated with customer i, $i \in N_c$, belongs the the discrete set of demands $\{g_{ivt} : v = \pi(i,t),\ldots,t-1\}$. Let a_{ivrt} be a binary coefficient that is equal to 1 if customer i belongs to route r of period t and has associated a demand equal to g_{ivt}. The formulation uses binary variable ξ_{rt} that is equal to 1 if route r for period t is in solution, 0 otherwise. The formulation is as follows:

$$
\begin{aligned}
(T2) \quad \min \quad & \sum_{t\in T}\Big(up_t + fy_t + h_0 I_{0t} + \sum_{r\in\hat{\mathcal{R}}_t} b_r\xi_{rt}\Big) + \sum_{t\in T'}\sum_{i\in N_c}\sum_{v=\pi(i,t)}^{t-1} e_{ivt}\lambda_{ivt} \\
\text{s.t.} \quad & (2)-(4), (6)-(8), (13), (35), (36), (37), \\
& \sum_{r\in\hat{\mathcal{R}}_t} a_{ivrt}\xi_{rt} = \lambda_{ivt}, \quad \forall i \in N_c,\ t \in T,\ v = \pi(i,t),\ldots,t-1, && (43) \\
& \sum_{r\in\hat{\mathcal{R}}_t} \xi_{rt} \leq m, \quad t \in T, && (44)
\end{aligned}
$$

$$0 \le y_t \le 1, \ \forall t \in T,$$

$$0 \le \lambda_{ivt} \le 1, \ \forall i \in N_c, v, t \in T',$$

$$\xi_{rt} \ge 0, \ \forall r \in \hat{\mathcal{R}}_t, t \in T.$$

Let $w_{ivt} \in \mathbb{R}$, $\forall i \in N_c$, $t \in T$, $v = \pi(i,t),\dots,t-1$, and $u_0 \le 0$ be the dual variables associated with constraints (43) and (44), respectively. The dual constraints associated with variables ξ_{rt} are

$$\sum_{i \in N_c} \sum_{v=\pi(i,t)}^{t-1} a_{ivrt} w_{ivt} + u_0 \le b_r, \quad \forall r \in \hat{\mathcal{R}}_t, t \in T,$$

and let (w^*, u_0^*) be the variables associated with a feasible dual solution of the above formulation. Based on Proposition 1, the following inequality provides a valid optimality cut:

$$\sum_{t \in T} \omega_t \ge \sum_{t \in T} \sum_{i \in N_c} \sum_{v=\pi(i,t)}^{t-1} w^*_{ivt} \lambda_{ivt} + m u_0^*. \tag{45}$$

4.2. Computing type I and II cuts

Optimality cuts (42) and (45) are derived by column generation based procedures to compute optimal dual solutions associated with formulations $T1$ and $T2$.

To speed up the computation, the route sets $\hat{\mathcal{R}}$ and $\hat{\mathcal{R}}_t, \forall t \in T$, of formulations $T1$ and $T2$, respectively, are extended with a relaxation of feasible routes easier to compute and based on the non-elementary route relaxation ng-route proposed by Baldacci et al. (2011), specifically tailored to consider variable demands associated with the customers. Let $O_i \subseteq N_c$ be a set of selected customers for vertex i such that $|O_i| = 8$, $\forall i \in N_c$, and O_i contains i and the seven nearest customers to i according to routing costs $\{c_{ij}\}$, and let t be a given time period. We define an ng-path (NG, q, i) as a nonnecessarily elementary path $P = (0, i_1, \dots, i_{k-1}, i_k = i)$ starting from the plant at time period t, visiting a subset of customers of total demand equal to q such that $NG = \Pi(P)$, ending at customer i, and such that $i \notin \Pi(P')$, where $P' = (0, i_1, \dots, i_{k-1})$. We denote by $\psi(NG, q, i)$ the cost of the least cost ng-path (NG, q, i) and we define an (NG, q, i)-route to be an $(NG, q, 0)$-path where i is the last customer visited before arriving at the plant. Functions $\psi(NG, q, i)$ can be computed using dynamic programming recursions similar to the recursions described by Baldacci et al. (2011) (details are omitted for sake of brevity) on a state space graph $\mathscr{H} = (\mathscr{E}, \Psi)$, where $\mathscr{E} = \{(NG, q, i) : g_{i\,t-1\,t} \le q \le Q, \forall NG \subseteq O_i \text{ s.t. } NG \ni i \text{ and } \sum_{j \in NG} g_{j\,t-1\,t} \le q, \forall i \in N\}$, $\Psi^{-1}(NG, q, i) = \{(NG', q - q_{it}, j) : \forall NG' \subseteq O_j \text{ s.t. } NG' \ni j \text{ and } NG' \cap O_i = NG \setminus \{i\}, \forall q_{it} \in \{g_{ivt} : v = \pi(i,t), \dots, t-1\}, \forall j \in N, j \ne i\}$, and $\Psi = \{(\Psi^{-1}(NG, q, i), (NG, q, i)) : \forall (NG, q, i) \in \mathscr{E}\}$.

5. Computational results

This section reports on the computational results of the exact method described in Section 3, hereafter called “EXM”. Experiments were conducted on both MVPRP and multi-vehicle VMIRP test instances generated by Adulyasak et al. (2014a) using the set of instances proposed by Archetti et al. (2007) and Archetti et al. (2011).

Method EXM was coded in Java and executed on a workstation equipped with an Intel(R) Xeon(R) CPU E5-2623 clocked at 3.00 GHz and 32 GB RAM, running under Linux operating system in a single-thread mode. ILOG CPLEX 12.6.3 (IBM CPLEX 2016) was used as the LP solver and the IP solver in EXM.

We compare EXM with the branch-and-cut methods of Adulyasak et al. (2014a) that, to the best of our knowledge, were the only to consider both the MVPRP and the multi-vehicle VMIRP under the OU policy. The experiments of Adulyasak et al. were performed on a workstation equipped with an Intel(R) Xeon(R) CPU clocked at 2.67 GHz and 24 GB RAM. According to the SuperPi (1M) benchmark (`http://www.superpi.net/`), an estimate of the single-thread speed of a CPU, our machine is about 10% faster than that used by Adulyasak et al..

The MVPRP test set consists of instances involving up to 50 customers with time horizons equal to 3, 6 and 9 time periods. The multi-vehicle VMIRP test set considers a number of customers ranging from 5 to 50 with time horizons equal to 3 and 6 periods. A total of 168 MVPRP instances and 320 multi-vehicle VMIRP instances have been considered in our experiments. For additional details about the instances considered in this section, the reader is referred to the the online supplement of Adulyasak et al.. The complete set of instances and the detailed results of Adulyasak et al. can be downloaded at the the website `https://sites.google.com/site/YossiriAdulyasak/publications`.

We compare method EXM with the following three versions of the branch-and-cut algorithm proposed by Adulyasak et al. (2014a):

- Veh-Ind: the vehicle index formulation running on a single core;
- Non-Veh-Ind: the nonvehicle index formulation running on a single core;
- 8c-Veh-Ind: the vehicle index formulation running on 8 cores.

For methods Veh-Ind and Non-Veh-Ind a time limit of 2 hours was imposed to the execution of the branch-and-cut algorithm whereas for 8c-Veh-Ind the maximum computing time was set to 12 hours of wall clock time. For method EXM, we imposed a time limit of 2 hours.

In the following, the results obtained on MVPRP instances are given in Section 5.1 whereas Section 5.2 reports the results obtained on multi-vehicle VMIRP instances. The results obtained by EXM can be downloaded at the website `http://www.computational-logistics.org/orlib/MVPRP`.

Table 1 Summary results on the MVPRP

			Veh-Ind			Non-Veh-Ind			8c-Veh-Ind				EXM			
l	n	$\#ist$	$\#opt$	t	$\%lb$	$\#opt$	t	$\%lb$	$\hat{n}$	$\#opt$	t	$\%lb$	$\hat{n}$	$\#opt$	t	$\%lb$
3	10-50	72	28	520.4	96.2	24	262.4	96.9	25	31	2493.2	96.5	40	44	1141.1	99.3
6	10-40	56	32	292.7	99.1	30	362.1	99.3	40	47	5531.5	98.3	40	53	489.0	98.5
9	10-30	40	19	970.8	98.2	13	739.1	99.1	30	28	4473.9	98.1	30	30	456.9	99.1
		168	79			67				106				127		

5.1. Computational results on the MVPRP

Table 1 summarises the results obtained on MVPRP instances. In the table, the instances are grouped according to the number of periods and columns "n" and "$\#ist$" report the ranges of the number of customers and the corresponding number of instances, respectively. For each method, Table 1 gives the number of instances solved to optimality within the imposed time limit ("$\#opt$"), the average computing time in seconds of the instances solved to optimality ("t") and the average percentage of the final lower bound relative to the best upper bound of the instances not solved to optimality ("$\%lb$"). For method 8c-Veh-Ind, the computing time refers to the wall clock time. For methods EXM and 8c-Veh-Ind the table also reports the number of customers of the instance solved to optimality ("$\hat{n}$") having the largest number of customers. The last row of the table reports the total number of instances solved to optimality by the different methods.

The results obtained can be summarized as follows:

- EXM solved to optimality 21 more instances than 8c-Veh-Ind, that represents the best method among the three versions proposed by Adulyasak et al. (2014a);
- Taking into account of the speed ratio between the machines used by the EXM and 8c-Veh-Ind and the fact that 8c-Veh-Ind also uses 8 cores, EXM is on average significantly faster than 8c-Veh-Ind;
- Instances with 40 customers were solved to optimality by EXM involving three periods, 15 customers more than the size of the instances that can be solved by 8c-Veh-Ind;
- The final lower bounds obtained by EXM for the instances not solved to optimality are on average quite tight.

Tables 2-4 show a detailed comparison of Veh-Ind, Non-Veh-Ind and EXM methods that were all run on a single core machine. The first four columns of the tables give details about the instances where column c reports the instance class. For the methods of Adulyasak et al. (2014a), the tables give the value of the best upper bound computed ("Best ub"), including method 8c-Veh-Ind, the percentage ratio of the lower bound relative to the best upper bound ("$\%lb$"), and the computing time in seconds ("t"). For method EXM, the tables give the cost of the best solution found ("ub"), the value of the final lower bound ("lb") and the corresponding percentage ratio relative to value ub ("$\%lb$"), the total computing time in seconds ("t"), the computing time in seconds spent in solving the subproblem ("t_S"), and the percentage ratio between the total routing cost and value ub ("$\%rc$"). Boldface numbers are used to indicate the best results among the upper bounds. In addition, a symbol "n/a" indicates that the corresponding results are not available. For the detailed results about method 8c-Veh-Ind the reader is referred to `https://sites.google.com/site/YossiriAdulyasak/publications`.

The detailed results on the whole set of instances can be summarized as follows:

- In terms of number of instances solved to optimality, EXM outperforms all the three versions proposed by Adulyasak et al. on instances with three and six periods. For instances with nine periods, EXM still outperforms method Non-Veh-Ind and cannot solve to optimality one and three instances solved by Veh-Ind and 8c-Veh-Ind methods, respectively;
- For 36 out of the 41 instances not solved to optimality by EXM, improved upper bounds were computed by EXM with respect to the upper bounds computed by 8c-Veh-Ind.
- Regarding the number of instances solved to optimality for each instance class, the tables show that 36, 38, 24 and 29 instances were solved to optimality for classes $c = 1, 2, 3, 4$, respectively. Therefore, EXM performs particularly well on instances of class 2, characterized by optimal solutions with a low ratio between the routing cost and the total cost.

5.1.1. Analysis of the different components of EXM. In this section, we analyze the impact of the different cuts embedded in method EXM. For sake of the comparison, we considered the 127 MVPRP instances solved to optimality by EXM, and we compare EXM with the following three versions of the method:

(A) Type I (42) and Type II (45) cuts described in Section 4.1 are not used;
(B) None of the two versions of the LP-based cuts (28) and (29) are added during the solution of the subproblem;
(C) The lifted cuts (27) and (29) are not used but their non-lifted versions (26) and (28) are used instead.

For the different versions, we used the same time limit and settings used for EXM.

Table 5 gives an overview of the results obtained. In the table, the instances are grouped according to the number of periods, and columns "*opt*" and "%*opt*" report the number of instances solved to optimality by the different versions and the corresponding percentage ratios computed over the total number of instances considered for each group of instances, respectively. The table then shows the number of optimality cuts (23) ("#cuts(23)"), the number of lifted infeasibility cuts (27) ("#cuts(27)"), the number of LP-based lifted optimality cuts (29) ("#cuts(29)"), and the number of executions of Step 5 of the algorithm used to solve problems $CVRP(t)$, i.e., the number of times CPLEX is invoked to solve the reduced problems ("$\#CVRP(t)$"). The next three columns report

Table 2 Detailed results on the MVPRP instances with three periods

				Adulyasak et al. (2014)					EXM					
					Veh-Ind		Non-Veh-Ind							
n	l	m	c	Best ub	$\%lb$	t	$\%lb$	t	ub	lb	$\%lb$	t	t_S	$\%rc$
10	3	2	1	36636	100.0	0.3	100.0	0.2	36636	36636.0	100.0	3.5	0.4	6.7
10	3	2	2	254526	100.0	0.1	100.0	0.0	254526	254526.0	100.0	3.2	0.0	1.0
10	3	2	3	46422	100.0	0.9	100.0	0.3	46422	46422.0	100.0	2.8	0.1	26.4
10	3	2	4	26687	100.0	0.3	100.0	0.1	26687	26687.0	100.0	3.0	0.0	9.2
10	3	3	1	37226	100.0	0.5	100.0	0.2	37226	37226.0	100.0	2.4	0.2	8.2
10	3	3	2	255116	100.0	0.2	100.0	0.1	255116	255116.0	100.0	2.3	0.0	1.2
10	3	3	3	49371	100.0	1.1	100.0	0.3	49371	49371.0	100.0	2.2	0.1	30.8
10	3	3	4	27247	100.0	0.5	100.0	0.1	27247	27247.0	100.0	2.3	0.0	11.1
15	3	2	1	56309	100.0	14.0	100.0	38.8	56309	56309.0	100.0	12.0	0.6	7.1
15	3	2	2	406122	100.0	14.7	100.0	20.5	406122	406122.0	100.0	12.6	0.8	1.2
15	3	2	3	71239	100.0	18.0	100.0	41.1	71239	71239.0	100.0	11.8	0.4	25.8
15	3	2	4	42978	100.0	23.7	100.0	116.4	42978	42978.0	100.0	15.2	2.9	9.2
15	3	3	1	57339	100.0	10.4	100.0	46.8	57339	57339.0	100.0	6.8	0.3	8.4
15	3	3	2	409891	100.0	11.0	100.0	187.0	409891	409891.0	100.0	8.3	0.8	1.4
15	3	3	3	76406	100.0	25.0	100.0	152.8	76406	76406.0	100.0	6.8	0.3	31.1
15	3	3	4	44293	100.0	24.0	100.0	551.0	44293	44293.0	100.0	7.8	0.7	10.7
20	3	2	1	57205	100.0	31.8	100.0	132.5	57205	57205.0	100.0	42.9	2.2	5.9
20	3	2	2	394852	100.0	107.3	100.0	63.0	394852	394852.0	100.0	46.0	0.9	0.9
20	3	2	3	69745	100.0	14.0	100.0	79.0	69745	69745.0	100.0	38.6	1.8	22.4
20	3	2	4	40893	100.0	85.3	100.0	91.6	40893	40893.0	100.0	46.4	3.2	7.6
20	3	3	1	57863	100.0	204.0	100.0	160.3	57863	57863.0	100.0	21.2	1.6	7.0
20	3	3	2	395363	100.0	185.9	100.0	90.7	395363	395363.0	100.0	20.4	1.1	1.0
20	3	3	3	74065	100.0	1700.4	100.0	4052.5	74065	74065.0	100.0	26.7	3.5	27.3
20	3	3	4	41550	100.0	679.7	100.0	471.5	41550	41550.0	100.0	23.9	3.1	9.7
25	3	2	1	78180	100.0	966.2	99.8	7200.0	78180	78180.0	100.0	182.5	21.5	5.4
25	3	2	2	564868	100.0	4639.4	100.0	7200.0	564868	564868.0	100.0	104.9	27.6	0.8
25	3	2	3	94420	100.0	1031.6	99.1	7200.0	94420	94420.0	100.0	549.4	30.0	21.3
25	3	2	4	58528	100.0	4782.1	99.2	7200.0	58528	58528.0	100.0	1696.3	114.5	7.5
25	3	3	1	79151	99.3	7200.0	99.6	7200.0	79151	79151.0	100.0	288.8	15.3	6.6
25	3	3	2	565800	99.7	7200.0	99.8	7200.0	565800	565800.0	100.0	48.9	7.4	1.1
25	3	3	3	99139	96.2	7200.0	97.6	7200.0	99139	99139.0	100.0	1067.2	25.1	25.1
25	3	3	4	59535	98.3	7200.0	98.4	7200.0	**59426**	59426.0	100.0	2737.2	69.5	8.9
30	3	3	1	82570	99.0	7200.0	99.0	7200.0	**82361**	82361.0	100.0	5088.8	435.8	6.0
30	3	3	2	586405	99.6	7200.0	99.7	7200.0	**585391**	585391.0	100.0	188.8	38.2	1.0
30	3	3	3	102234	95.5	7200.0	95.9	7200.0	**101707**	101197.9	99.5	7200.3	590.5	23.4
30	3	3	4	61945	97.1	7200.0	97.4	7200.0	**61010**	60924.7	99.9	7200.8	1203.1	8.1
30	3	4	1	83738	98.1	7200.0	98.5	7200.0	**83316**	83284.1	100.0	7200.3	292.5	7.1
30	3	4	2	587571	99.6	7200.0	99.7	7200.0	**586933**	586933.0	100.0	205.3	102.1	1.1
30	3	4	3	109845	91.4	7200.0	93.3	7200.0	**106835**	105952.8	99.2	7200.3	1043.5	27.1
30	3	4	4	63156	96.4	7200.0	96.8	7200.0	**62145**	61961.8	99.7	7200.3	1651.6	9.5
35	3	3	1	96528	96.6	7200.0	96.8	7200.0	**94349**	94349.0	100.0	6840.5	756.6	5.9
35	3	3	2	661386	99.4	7200.0	99.5	7200.0	**658885**	658885.0	100.0	339.3	101.2	0.9
35	3	3	3	122256	90.4	7200.0	91.1	7200.0	**116478**	115231.3	98.9	7200.2	756.0	23.5
35	3	3	4	70948	96.3	7200.0	96.5	7200.0	**69440**	69440.0	100.0	6163.5	929.4	7.9
35	3	4	1	98239	95.4	7200.0	96.0	7200.0	**95296**	95296.0	100.0	6049.2	785.9	6.9
35	3	4	2	661447	99.5	7200.0	99.6	7200.0	**660203**	660203.0	100.0	317.5	171.5	1.2
35	3	4	3	128029	88.8	7200.0	90.6	7200.0	**121137**	121137.0	100.0	6716.9	833.4	26.6
35	3	4	4	72571	94.9	7200.0	95.5	7200.0	**70467**	70307.6	99.8	7200.2	1090.8	9.1
40	3	3	1	127280	98.8	7200.0	98.8	7200.0	**126821**	126821.0	100.0	5148.8	340.4	5.0
40	3	3	2	898014	99.6	7200.0	99.6	7200.0	**896020**	896020.0	100.0	3899.1	1160.1	0.8
40	3	3	3	153029	94.8	7200.0	95.1	7200.0	**151526**	149860.7	98.9	7200.2	1676.2	19.8
40	3	3	4	92922	97.3	7200.0	97.5	7200.0	**91929**	91428.7	99.5	7200.3	2215.0	6.9
40	3	4	1	129067	97.8	7200.0	98.1	7200.0	**128157**	128034.2	99.9	7200.3	544.1	6.0
40	3	4	2	901594	99.2	7200.0	99.3	7200.0	**897145**	897145.0	100.0	2207.4	900.3	0.9
40	3	4	3	164191	89.5	7200.0	90.6	7200.0	**158049**	156334.9	98.9	7200.3	2107.6	23.2
40	3	4	4	95927	94.7	7200.0	95.2	7200.0	**93097**	92761.8	99.6	7200.3	1895.6	8.0
45	3	3	1	141856	98.3	7200.0	98.6	7200.0	**141265**	141075.0	99.9	7200.3	571.8	5.0
45	3	3	2	1025062	99.3	7200.0	99.4	7200.0	**1020302**	1019985.0	100.0	7200.3	2611.2	0.8
45	3	3	3	172851	92.9	7200.0	93.2	7200.0	**169147**	167377.4	99.0	7200.3	2627.6	19.7
45	3	3	4	106750	96.5	7200.0	96.8	7200.0	**105017**	104445.3	99.5	7200.2	2332.9	6.9
45	3	4	1	144963	96.8	7200.0	97.1	7200.0	**142958**	142507.3	99.7	7200.3	1433.0	6.2
45	3	4	2	1027296	99.1	7200.0	99.2	7200.0	**1021791**	1021528.5	100.0	7200.3	3337.9	0.9
45	3	4	3	183037	88.7	7200.0	90.4	7200.0	**176584**	174930.9	99.1	7200.3	1607.8	23.2
45	3	4	4	108419	95.5	7200.0	96.1	7200.0	**106684**	106047.1	99.4	7200.3	2428.5	8.1
50	3	3	1	139164	98.0	7200.0	98.1	7200.0	**138235**	137491.7	99.5	7200.3	5122.7	5.8
50	3	3	2	980022	99.4	7200.0	99.4	7200.0	**976479**	975639.3	99.9	7200.3	6509.8	0.8
50	3	3	3	173118	91.4	7200.0	92.1	7200.0	**172256**	164668.0	95.6	7203.7	6839.8	23.4
50	3	3	4	103031	96.1	7200.0	96.3	7200.0	**100972**	100239.7	99.3	7200.2	3985.2	7.5
50	3	4	1	141178	97.2	7200.0	97.5	7200.0	**139950**	139143.0	99.4	7200.2	2340.3	6.8
50	3	4	2	983352	99.2	7200.0	99.2	7200.0	**978174**	977274.8	99.9	7200.3	2319.0	1.0
50	3	4	3	189474	85.9	7200.0	87.2	7200.0	**178483**	173194.1	97.0	7201.6	6875.9	26.4
50	3	4	4	104222	95.8	7200.0	96.3	7200.0	**102748**	102007.8	99.3	7200.2	2467.8	9.1

Table 3 Detailed results on the MVPRP instances with six periods

				Adulyasak et al. (2014)					EXM					
					Veh-Ind		Non-Veh-Ind							
n	l	m	c	Best ub	$\%lb$	t	$\%lb$	t	ub	lb	$\%lb$	t	t_S	$\%rc$
10	6	2	1	38669	100.0	0.4	100.0	0.2	38669	38669.0	100.0	5.7	0.1	7.3
10	6	2	2	222269	100.0	0.4	100.0	0.2	222269	222269.0	100.0	5.8	0.1	1.3
10	6	2	3	50025	100.0	0.5	100.0	0.2	50025	50025.0	100.0	6.6	0.2	28.4
10	6	2	4	23453	100.0	0.7	100.0	0.4	23453	23453.0	100.0	6.6	0.9	12.1
10	6	3	1	38856	100.0	0.6	100.0	0.2	38856	38856.0	100.0	4.4	0.1	7.8
10	6	3	2	222456	100.0	1.2	100.0	0.1	222456	222456.0	100.0	4.5	0.1	1.4
10	6	3	3	50963	100.0	1.4	100.0	0.2	50963	50963.0	100.0	4.4	0.1	29.7
10	6	3	4	23640	100.0	2.1	100.0	0.3	23640	23640.0	100.0	4.7	0.1	12.8
15	6	2	1	54845	100.0	5.7	100.0	2.3	54845	54845.0	100.0	25.7	1.6	7.8
15	6	2	2	307565	100.0	5.3	100.0	2.3	307565	307565.0	100.0	26.0	1.6	1.4
15	6	2	3	71661	100.0	9.9	100.0	44.9	71661	71661.0	100.0	28.0	1.0	27.1
15	6	2	4	32475	100.0	13.4	100.0	91.8	32475	32475.0	100.0	33.1	6.1	12.7
15	6	3	1	55726	100.0	27.9	100.0	10.9	55726	55726.0	100.0	12.4	1.2	9.2
15	6	3	2	308446	100.0	28.1	100.0	13.7	308446	308446.0	100.0	12.7	1.7	1.7
15	6	3	3	75004	100.0	150.3	100.0	176.3	75004	75004.0	100.0	16.2	3.6	31.0
15	6	3	4	33178	100.0	89.0	100.0	386.4	33178	33178.0	100.0	17.2	5.1	14.7
20	6	2	1	64447	100.0	25.7	100.0	120.1	64447	64447.0	100.0	146.2	2.9	6.6
20	6	2	2	361987	100.0	16.4	100.0	26.7	361987	361987.0	100.0	152.8	3.1	1.2
20	6	2	3	80568	100.0	38.3	100.0	483.4	80568	80568.0	100.0	174.4	5.3	24.8
20	6	2	4	37798	100.0	14.2	100.0	496.2	37798	37798.0	100.0	151.0	2.0	10.6
20	6	3	1	65111	100.0	304.7	100.0	656.6	65111	65111.0	100.0	72.0	8.0	7.6
20	6	3	2	362651	100.0	151.4	100.0	23.0	362651	362651.0	100.0	69.7	6.0	1.4
20	6	3	3	83347	100.0	408.5	100.0	417.8	83347	83347.0	100.0	62.0	3.2	27.3
20	6	3	4	38355	100.0	214.1	100.0	91.2	38355	38355.0	100.0	64.5	1.8	11.8
25	6	2	1	80401	100.0	425.9	100.0	2160.3	80401	80401.0	100.0	208.2	10.3	6.1
25	6	2	2	430861	100.0	244.4	100.0	880.3	430861	430861.0	100.0	199.8	8.6	1.1
25	6	2	3	99385	100.0	445.0	100.0	988.0	99385	99385.0	100.0	288.7	31.3	23.7
25	6	2	4	45070	100.0	199.8	100.0	2180.9	45070	45070.0	100.0	220.2	14.3	10.4
25	6	3	1	81155	99.7	7200.0	99.6	7200.0	81155	81155.0	100.0	106.6	21.9	7.1
25	6	3	2	431615	100.0	7200.0	99.9	7200.0	431615	431615.0	100.0	100.3	18.4	1.3
25	6	3	3	102924	98.1	7200.0	98.8	7200.0	102924	102924.0	100.0	194.0	28.8	25.1
25	6	3	4	45743	99.4	7200.0	99.3	7200.0	45743	45743.0	100.0	105.5	10.3	12.0
30	6	3	1	81067	100.0	651.9	100.0	621.2	81067	81067.0	100.0	579.1	160.9	6.9
30	6	3	2	458257	100.0	775.6	100.0	987.7	458257	458257.0	100.0	462.1	48.2	1.2
30	6	3	3	102824	100.0	3683.1	99.1	7200.0	102824	102824.0	100.0	1500.3	964.9	26.0
30	6	3	4	47649	100.0	1429.1	99.7	7200.0	47649	47649.0	100.0	611.8	123.2	11.2
30	6	4	1	81697	99.8	7200.0	99.8	7200.0	81697	81697.0	100.0	219.0	18.6	7.6
30	6	4	2	458887	100.0	7200.0	100.0	7200.0	458887	458887.0	100.0	208.3	15.3	1.3
30	6	4	3	106086	97.0	7200.0	97.9	7200.0	106086	106086.0	100.0	456.7	120.5	28.4
30	6	4	4	48296	99.0	7200.0	99.2	7200.0	48296	48296.0	100.0	310.7	54.7	12.6
35	6	3	1	99205	n/a	n/a	n/a	n/a	99205	99205.0	100.0	653.4	34.2	6.2
35	6	3	2	570355	n/a	n/a	n/a	n/a	570355	570355.0	100.0	695.8	42.0	1.1
35	6	3	3	123688	n/a	n/a	n/a	n/a	123688	123688.0	100.0	1324.3	561.3	24.2
35	6	3	4	59046	n/a	n/a	n/a	n/a	59046	59046.0	100.0	931.3	219.2	10.1
35	6	4	1	100225	n/a	n/a	n/a	n/a	100225	100225.0	100.0	513.3	242.8	7.1
35	6	4	2	571385	n/a	n/a	n/a	n/a	**571375**	571375.0	100.0	607.6	335.1	1.3
35	6	4	3	**129846**	n/a	n/a	n/a	n/a	129922	126773.6	97.6	7201.1	6925.8	27.9
35	6	4	4	59913	n/a	n/a	n/a	n/a	**59878**	59878.0	100.0	3084.2	1144.0	11.6
40	6	3	1	133248	n/a	n/a	n/a	n/a	133248	133248.0	100.0	1695.5	815.4	5.0
40	6	3	2	734268	n/a	n/a	n/a	n/a	734268	734268.0	100.0	1295.8	455.0	0.9
40	6	3	3	**160896**	n/a	n/a	n/a	n/a	162069	159057.1	98.1	7206.3	6483.8	21.7
40	6	3	4	74693	n/a	n/a	n/a	n/a	**74621**	74621.0	100.0	6284.7	4519.1	9.4
40	6	4	1	135077	n/a	n/a	n/a	n/a	**134281**	134281.0	100.0	431.5	79.6	5.7
40	6	4	2	736464	n/a	n/a	n/a	n/a	**735301**	735301.0	100.0	485.6	112.4	1.0
40	6	4	3	168262	n/a	n/a	n/a	n/a	**164762**	164626.6	99.9	7200.3	435.5	23.0
40	6	4	4	76062	n/a	n/a	n/a	n/a	**75505**	75505.0	100.0	1037.9	459.2	10.6

Table 4 Detailed results on the MVPRP instances with nine periods

				Adulyasak et al. (2014)					EXM					
					Veh-Ind		Non-Veh-Ind							
n	l	m	c	Best ub	$\%lb$	t	$\%lb$	t	ub	lb	$\%lb$	t	t_S	$\%rc$
10	9	2	1	63064	100.0	4.3	100.0	1.9	63064	63064.0	100.0	11.7	1.6	7.9
10	9	2	2	381394	100.0	5.3	100.0	2.1	381394	381394.0	100.0	11.9	1.8	1.3
10	9	2	3	82683	100.0	31.7	100.0	8.6	82683	82683.0	100.0	16.1	1.9	29.3
10	9	2	4	40774	100.0	19.9	100.0	5.6	40774	40774.0	100.0	15.9	2.3	12.9
10	9	3	1	63822	100.0	18.6	100.0	20.2	63822	63822.0	100.0	6.9	0.4	9.2
10	9	3	2	382152	100.0	31.9	100.0	21.6	382152	382152.0	100.0	6.6	0.5	1.5
10	9	3	3	86095	100.0	124.3	100.0	58.4	86095	86095.0	100.0	11.6	1.3	32.2
10	9	3	4	41379	100.0	126.3	100.0	41.9	41379	41379.0	100.0	10.2	1.7	13.3
15	9	2	1	91148	100.0	118.2	100.0	3440.1	91148	91148.0	100.0	49.7	1.6	7.6
15	9	2	2	540698	100.0	295.9	100.0	2734.2	540698	540698.0	100.0	46.8	1.6	1.3
15	9	2	3	118746	100.0	2160.8	98.4	7200.0	118746	118746.0	100.0	470.6	6.9	28.0
15	9	2	4	57753	100.0	973.7	99.3	7200.0	57753	57753.0	100.0	126.1	4.7	12.0
15	9	3	1	92632	100.0	3112.6	99.7	7200.0	92632	92632.0	100.0	23.2	2.5	9.1
15	9	3	2	542182	100.0	4935.5	100.0	7200.0	542182	542182.0	100.0	22.5	1.9	1.5
15	9	3	3	125383	95.7	7200.0	97.6	7200.0	125383	125383.0	100.0	327.2	6.1	32.1
15	9	3	4	59386	98.5	7200.0	98.5	7200.0	59386	59386.0	100.0	1935.6	21.4	13.9
20	9	2	1	103809	100.0	833.3	100.0	1854.9	103809	103809.0	100.0	306.1	8.2	7.0
20	9	2	2	617889	100.0	153.9	100.0	70.2	617889	617889.0	100.0	396.3	3.1	1.2
20	9	2	3	131101	100.0	792.5	99.2	7200.0	131101	130587.8	99.6	7200.3	347.0	25.3
20	9	2	4	65859	100.0	583.7	99.7	7200.0	65859	65859.0	100.0	502.2	95.0	12.2
20	9	3	1	104704	99.7	7200.0	99.9	7200.0	104704	104704.0	100.0	138.2	4.6	8.4
20	9	3	2	618902	100.0	4121.9	100.0	1348.8	618902	618902.0	100.0	135.4	13.0	1.4
20	9	3	3	136443	97.3	7200.0	97.9	7200.0	**136286**	135522.4	99.4	7200.2	42.1	28.5
20	9	3	4	66830	99.6	7200.0	99.4	7200.0	66830	66830.0	100.0	575.5	59.7	13.5
25	9	2	1	129172	n/a	n/a	n/a	n/a	129172	129172.0	100.0	388.6	18.1	5.8
25	9	2	2	749311	n/a	n/a	n/a	n/a	749311	749311.0	100.0	364.0	10.5	1.1
25	9	2	3	**158573**	n/a	n/a	n/a	n/a	158684	157712.2	99.4	7200.3	768.4	22.9
25	9	2	4	**79476**	n/a	n/a	n/a	n/a	79496	79300.9	99.8	7200.3	216.0	10.6
25	9	3	1	130594	n/a	n/a	n/a	n/a	**130550**	130550.0	100.0	515.3	14.6	7.0
25	9	3	2	750481	n/a	n/a	n/a	n/a	750481	750481.0	100.0	166.6	14.3	1.2
25	9	3	3	167271	n/a	n/a	n/a	n/a	**165430**	163824.1	99.0	7200.3	329.6	25.4
25	9	3	4	81012	n/a	n/a	n/a	n/a	**80761**	80272.2	99.4	7200.3	128.1	11.8
30	9	3	1	137463	n/a	n/a	n/a	n/a	137463	137463.0	100.0	931.8	29.3	6.3
30	9	3	2	828543	n/a	n/a	n/a	n/a	**828529**	828529.0	100.0	2958.5	331.8	1.2
30	9	3	3	174270	n/a	n/a	n/a	n/a	**173868**	171090.1	98.4	7200.6	722.7	25.7
30	9	3	4	87607	n/a	n/a	n/a	n/a	**87456**	86875.7	99.3	7200.2	567.2	11.1
30	9	4	1	139909	n/a	n/a	n/a	n/a	**138887**	138887.0	100.0	1906.6	143.5	7.4
30	9	4	2	830800	n/a	n/a	n/a	n/a	**829697**	829697.0	100.0	1328.2	174.1	1.3
30	9	4	3	182839	n/a	n/a	n/a	n/a	**181650**	177116.8	97.5	7200.5	348.8	27.9
30	9	4	4	89684	n/a	n/a	n/a	n/a	**88541**	88043.4	99.4	7200.3	447.1	12.3

computing times about the solution of the BPPs used to generate infeasibility cuts ("t_{BPP}"), the heuristic used to compute the upper bounds z_{UB} in solving problems $CVRP(t)$ ("t_{heu}"), and the time spent in solving the subproblem ("t_{sub}"). The last column reports the total computing time ("t_{tot}"). Regarding the number of cuts and the computing times, the table reports average values, and the last line of each table section dedicated to a version, reports the total number of instances solved to optimality and average values over the different columns. Moreover, for version (C) of EXM, the table reports the average numbers of non-lifted cuts added instead of the average number of lifted cuts.

Table 5 Analysis of the different components of EXM on the MVPRP

Version	l	$\#inst$	opt	$\%opt$	#cuts(23)	#cuts(27)	#cuts(29)	$\#CVRP(t)$	t_{BPP}	t_{heu}	t_{sub}	t_{tot}
EXM	3	44	44	100.0	13.5	234.8	1009.4	4.8	55.6	2.4	156.6	1141.1
	6	53	53	100.0	253.6	2.9	425.0	21.2	4.2	1.6	196.8	489.0
	9	30	30	100.0	245.4	27.1	730.6	20.3	5.7	1.0	32.6	456.9
			127		168.5	89.0	699.7	15.3	22.4	1.7	144.1	702.9
A (no type I and II cuts)	3	44	39	88.6	15.1	353.5	1729.2	4.3	106.8	3.8	191.5	1450.5
	6	53	51	96.2	183.9	3.3	418.0	23.5	3.9	1.2	491.7	543.0
	9	30	27	90.0	328.0	35.0	895.0	26.0	9.0	1.3	37.8	1034.8
			117		159.5	132.1	985.0	17.4	40.8	2.2	280.5	973.6
B (no optimality cuts (28), (29))	3	44	26	59.1	6664.7	569.0	-	2096.1	182.4	10.9	1059.1	3183.4
	6	53	32	60.4	2592.4	6.1	-	915.7	19.0	3.3	2401.9	3315.7
	9	30	6	20.0	9975.1	91.6	-	4510.4	96.8	8.9	968.7	5814.1
			64		5747.2	221.3		2173.8	94.0	7.2	1598.1	3860.0
						#cuts(26)	#cuts(28)					
C (no lifted cuts (27), (29))	3	44	40	90.9	13.2	232.3	994.6	4.1	52.6	2.4	105.1	1168.7
	6	53	51	96.2	287.7	3.1	677.8	26.9	5.8	2.3	242.6	728.7
	9	30	25	83.3	401.3	35.6	1280.6	31.2	10.3	1.7	53.6	1752.7
			116		219.4	90.2	929.9	20.1	23.1	2.2	150.3	1123.0

The results obtained can be analyzed as follows.

- The initialization of EXM using Type I and Type II cuts and the use of the lifted versions of infeasibility and LP-based cuts are quite effective, since versions (A) and (C) cannot solve to optimality 10 and 11 instances solved by EXM, respectively.
- EXM takes particularly advantage in using LP-based cuts (29). Indeed, version (B) solved to optimality only 64 instances over the 127 instances considered. As shown by Table 5, if the LP-based cuts are not used, the number of optimality cuts (23) and the number of times the IP solver of CPLEX is invoked (and the corresponding average computing times) increase considerately.

In summary, all the different cuts embedded in EXM are particulary effective in solving MVPRP instances.

Concerning version (A) of EXM, we also executed EXM by selectively disabling the use of Type I and Type II cuts. As a result, EXM initialized with only Type I cut solved to optimality 116 instances whereas EXM with only Type II cut solved 117 instances. Finally, the percentage ratio of lower bound LB_R on the routing cost used in the definition of the optimality cuts (23), computed using the routing costs of the 127 instances considered, is equal to 55.0%, hence resulting particularly weak. An explanation for its quality is due to the definition of values $\{q_i\}$, i.e., the lower bounds on the quantity to be delivered. Nevertheless, its computing time is negligible, being on average equal to 9.2 seconds, and our aim in computing LB_R was to quickly compute an initial lower bound to properly define the optimality cuts (23).

Table 6 Summary results on the multi-vehicle VMIRP

				Veh-Ind				Non-Veh-Ind				8c-Veh-Ind				EXM			
l	class	n	$\#ist$	$\hat{n}$	$\#opt$	t	$\%lb$	$\hat{n}$	$\#opt$	t	$\%lb$	$\hat{n}$	$\#opt$	t	$\%lb$	$\hat{n}$	$\#opt$	t	$\%lb$
3	Low	5-50	100	40	59	670.4	89.5	30	45	530.2	94.6	50	78	4279.1	88.9	30	50	892.5	86.5
3	High	5-50	100	40	60	745.0	95.8	40	47	808.1	97.6	50	77	3599.7	95.4	30	47	668.0	95.1
6	Low	5-30	60	15	24	671.3	93.8	10	16	622.8	94.8	25	37	4610.9	87.2	10	20	358.7	91.4
6	High	5-30	60	15	24	658.1	96.4	10	16	617.2	96.9	25	37	4651.1	93.1	10	20	326.8	95.0
			320		167				124				229				137		

5.2. Computational results on the multi-vehicle VMIRP

In this section, we report on the results obtained by EXM on multi-vehicle VMIRP, a special case of the MVPRP defined as follows:

- The fixed production setup cost f and the unit production cost u are set equal to 0;
- All variables y are set equal to 1, i.e., $y_t = 1$, $\forall t \in T$;
- The production quantity in period t is fixed to B_t, i.e., $p_t = B_t$, $\forall t \in T$, where B_t is the production quantity made available in each period. The additional constraints $I_{ot} \geq B_t$, $\forall t \in T$, are added to formulation F and the term $\sum_{i \in N_c} h_i I_{i0}$ is added to the objective function of F.

As assumed by Adulyasak et al. (2014a), the production at the plant takes place before delivery and the deliveries at the customers are executed at the beginning of the time period.

Table 6 summarizes the results obtained whereas tables (7)-(10) give the details about methods Veh-Ind, Non-Veh-Ind and EXM. The notation used in the tables of this section is as described in the previous section about the MVPRP. Moreover, in Table 6 the instances are grouped according to the type of inventory costs (*Low* or *Hight*) and in tables (7)-(10) column "*id*" is used to denote the instance identifier. Concerning method EXM, a symbol "-" indicates that EXM runs out of memory or no feasible solution was found by EXM.

We run EXM on the whole set of multi-vehicle VMIRP instances generated by Adulyasak et al.. In particular, we also considered 20 instances involving 30 customers and six periods (with both low and hight inventory costs) that were not run by Adulyasak et al. with any of their branch-and-cut versions.

The results about the multi-vehicle VMIRP can be analysed as follows.

- In terms of the number of instances solved to optimality, EXM outperforms method Non-Veh-Ind but it is outperformed by the methods based on the vehicle index formulation (Veh-Ind and 8c-Veh-Ind).
- The detailed results show that EXM is not dominated by method Veh-Ind as it can solve four instances to optimality that were not solved by Veh-Ind within the imposed time limit. In addition, EXM computed 20 new improved upper bounds with respect to the best upper bounds computed by method 8c-Veh-Ind.

- EXM scales particularly well with the number of periods. Indeed, new upper and lower bounds for the instances with 30 customers and six periods were computed by EXM.

The detailed results about the MVPRP and the multi-vehicle VMIRP show that the average ratios between the routing cost and the total cost of the best solutions obtained (column "$\%rc$") are equal to 11.3% and 66.7%, respectively. This is due to the cost structure of the instances and, in particular, to the fact that in the multi-vehicle VMIRP setup and production costs are not considered. The results obtained show that EXM is particular efficient on MVPRP instances. A possible explanation for this behaviour is due to the fact that, since setup, production and inventory costs dominate the routing cost, the master problem provides a tight lower bound on the optimal solution cost, thus speeding up the convergence of EXM.

6. Conclusions and future research

We presented an exact Benders decomposition algorithm for the Multi-Vehicle Production Routing Problem (MVPRP) with Order-Up-to level inventory replenishment policy and for its special case, the multi-vehicle Vendor-Managed Inventory Routing Problem (VMIRP).

We demonstrated through extensive computational experiments that our approach outperforms the state-of-the-art method for the MVPRP. In particular, the method could solve to optimality MVPRP instances with up to 40 customers, three periods, and three vehicles, that were not solved to optimality by the state-of-the-art method. Moreover, our approach is also competitive on multi-vehicle VMIRP instances and could compute new lower and upper bounds for difficult multi-vehicle VMIRP instances with up to 30 customers, six periods, and four vehicles.

The proposed method, due to its decomposition structure, can be easily adapted to deal with other intra-route constraints, simply by taking into account of such constraints in the route generation phase. Future research will therefore investigate the generalization of the model described in this paper to deal with the complexity of real-world production routing applications, such as time windows, distance constraints, etc.

Acknowledgments

The authors would like to thank the anonymous reviewers and associate editor for their helpful suggestions and very thorough review of the paper.

This research was partially supported by the National Natural Science Foundation of China (Grant Nos. 71501091, 71571094) and NRF Singapore [Grant NRFRSS2016-004], and MOE-AcRF-Tier 1 [Grants R-266-000-096-133, R-266-000-096-731, and R-266-000-100-646] and Tier 2 [Grant MOE2017-T2-2-153].

Table 7 Results on multi-vehicle VMIRP instances with three periods and low inventory costs

					Adulyasak et al. (2014)				EXM					
					Veh-Ind		Non-Veh-Ind							
n	l	m	id	Best ub	$\%lb$	t	$\%lb$	t	ub	lb	$\%lb$	t	t_S	$\%rc$
5	3	2	1	1396.33	100.0	0.1	100.0	0.1	1396.33	1396.33	100.0	0.7	0.0	93.2
5	3	2	2	1437.90	100.0	0.0	100.0	0.0	1437.90	1437.90	100.0	0.7	0.0	93.8
5	3	2	3	2468.94	100.0	0.1	100.0	0.3	2468.94	2468.94	100.0	0.8	0.0	94.4
5	3	2	4	1717.43	100.0	0.0	100.0	0.1	1717.43	1717.43	100.0	0.7	0.0	96.2
5	3	2	5	1224.63	100.0	0.0	100.0	0.0	1224.63	1224.63	100.0	0.7	0.0	89.1
5	3	3	1	1540.57	100.0	0.1	100.0	0.1	1540.57	1540.57	100.0	0.6	0.0	93.9
5	3	3	2	1720.83	100.0	0.1	100.0	0.1	1720.83	1720.83	100.0	0.5	0.0	94.8
5	3	3	3	3503.33	100.0	0.0	100.0	0.1	3503.33	3503.33	100.0	0.7	0.0	96.0
5	3	3	4	2552.79	100.0	0.0	100.0	0.1	2552.79	2552.79	100.0	0.5	0.0	97.5
5	3	3	5	1682.44	100.0	0.0	100.0	0.1	1682.44	1682.44	100.0	0.6	0.0	92.0
10	3	2	1	2468.22	100.0	1.8	100.0	1.7	2468.22	2468.22	100.0	4.6	1.5	87.6
10	3	2	2	3166.19	100.0	2.9	100.0	3.5	3166.19	3166.19	100.0	6.4	2.9	91.8
10	3	2	3	2449.10	100.0	1.2	100.0	1.6	2449.10	2449.10	100.0	3.4	0.5	90.1
10	3	2	4	2859.45	100.0	2.7	100.0	5.4	2859.45	2859.45	100.0	5.8	3.0	91.4
10	3	2	5	2486.41	100.0	1.6	100.0	2.6	2486.41	2486.41	100.0	5.3	2.2	87.2
10	3	3	1	2914.09	100.0	4.1	100.0	5.2	2914.09	2914.09	100.0	2.6	0.6	89.5
10	3	3	2	3641.19	100.0	7.1	100.0	3.3	3641.19	3641.19	100.0	2.4	0.5	92.9
10	3	3	3	2734.10	100.0	2.4	100.0	1.3	2734.10	2734.10	100.0	2.0	0.2	91.1
10	3	3	4	3318.99	100.0	7.2	100.0	10.8	3318.99	3318.99	100.0	4.2	2.2	92.6
10	3	3	5	2704.71	100.0	3.3	100.0	1.5	2704.71	2704.71	100.0	3.0	0.6	88.3
15	3	2	1	2631.53	100.0	7.3	100.0	8.5	2631.53	2631.53	100.0	19.8	2.4	85.5
15	3	2	2	2907.34	100.0	5.2	100.0	36.5	2907.34	2907.34	100.0	27.5	5.9	87.4
15	3	2	3	3081.56	100.0	4.9	100.0	5.3	3081.56	3081.56	100.0	18.0	1.4	86.1
15	3	2	4	2745.98	100.0	3.6	100.0	24.3	2745.98	2746.98	100.0	23.1	6.1	88.6
15	3	2	5	2862.50	100.0	6.1	100.0	58.9	2862.50	2862.50	100.0	24.5	6.4	89.2
15	3	3	1	2928.16	100.0	12.5	100.0	109.3	2928.16	2928.16	100.0	9.7	1.6	87.1
15	3	3	2	3217.61	100.0	17.3	100.0	88.2	3217.61	3217.61	100.0	19.0	6.3	88.5
15	3	3	3	3401.56	100.0	20.3	100.0	36.3	3401.56	3401.56	100.0	10.7	1.9	87.4
15	3	3	4	2961.29	100.0	10.6	100.0	12.2	2961.29	2961.29	100.0	12.2	3.2	89.5
15	3	3	5	3302.02	100.0	35.0	100.0	170.4	3302.02	3302.02	100.0	17.7	6.3	90.6
20	3	2	1	3398.29	100.0	128.1	100.0	278.4	3398.29	3398.29	100.0	364.0	123.5	85.2
20	3	2	2	2925.55	100.0	18.9	100.0	21.9	2925.55	2925.55	100.0	105.6	29.0	82.4
20	3	2	3	3401.56	100.0	50.1	100.0	128.0	3401.56	3401.56	100.0	117.6	28.7	84.1
20	3	2	4	3837.54	100.0	64.7	99.2	7200.0	3837.54	3837.54	100.0	243.5	71.8	89.0
20	3	2	5	3957.49	100.0	73.9	100.0	1331.3	3957.49	3957.49	100.0	1998.8	356.8	85.7
20	3	3	1	3809.57	100.0	337.3	100.0	1552.4	3809.57	3809.57	100.0	976.4	94.5	86.7
20	3	3	2	3063.77	100.0	67.4	100.0	95.0	3063.77	3063.77	100.0	49.0	12.6	83.2
20	3	3	3	3805.55	100.0	359.3	100.0	5953.2	3805.55	3805.55	100.0	309.6	46.0	85.8
20	3	3	4	4447.93	100.0	899.0	95.4	7200.0	4447.93	4447.93	100.0	1697.7	147.0	90.4
20	3	3	5	4581.99	100.0	882.3	97.4	7200.0	4581.99	4581.99	100.0	5524.8	242.8	87.7
25	3	2	1	3569.23	100.0	25.7	100.0	164.8	3569.23	3569.23	100.0	5061.4	4598.9	83.4
25	3	2	2	3904.97	100.0	447.8	100.0	1798.3	3904.97	3904.97	100.0	2678.3	1253.8	83.7
25	3	2	3	4006.11	100.0	93.3	100.0	434.5	4006.11	4006.11	100.0	1332.8	984.2	82.2
25	3	2	4	3510.77	100.0	32.6	100.0	100.5	3510.77	3510.77	100.0	5254.6	5015.2	82.8
25	3	2	5	3857.48	100.0	39.5	100.0	20.0	3857.48	3857.48	100.0	608.6	194.5	79.3
25	3	3	1	3927.38	100.0	219.8	100.0	1521.0	3927.38	3927.38	100.0	3224.5	262.4	84.9
25	3	3	2	4278.43	100.0	1609.5	97.0	7200.0	4278.43	4278.43	100.0	5016.6	524.7	85.1
25	3	3	3	4658.61	100.0	1890.2	98.9	7200.0	4658.61	4613.86	99.0	7200.3	370.4	84.7
25	3	3	4	3936.55	100.0	238.8	98.3	7200.0	3936.55	3936.55	100.0	697.4	171.0	84.7
25	3	3	5	4550.44	100.0	1218.0	100.0	5702.7	4550.44	4392.76	96.5	7200.3	2061.4	82.5
30	3	3	1	**4697.49**	100.0	2956.4	97.1	7200.0	-	4623.26	-	7201.7	7037.8	-
30	3	3	2	4296.11	100.0	334.6	100.0	4168.2	4296.11	4187.66	97.5	7200.3	1420.9	80.3
30	3	3	3	4272.41	100.0	2256.8	98.7	7200.0	4272.41	4272.41	100.0	2995.8	373.7	77.1
30	3	3	4	4238.99	100.0	4683.7	95.7	7200.0	4238.99	4139.71	97.7	7200.3	2951.0	83.2
30	3	3	5	3987.55	100.0	5038.9	93.8	7200.0	3987.55	3962.85	99.4	7200.3	794.2	80.7
30	3	4	1	**5297.66**	89.1	7200.0	92.7	7200.0	5318.99	5136.46	96.6	7200.3	1976.9	81.9
30	3	4	2	4634.58	100.0	4129.9	97.4	7200.0	4634.58	4614.09	99.6	7200.3	1575.1	81.8
30	3	4	3	4569.05	92.3	7200.0	95.6	7200.0	4569.05	4544.04	99.5	7200.3	221.7	78.6
30	3	4	4	**4701.86**	87.2	7200.0	93.8	7200.0	4915.80	4595.86	93.5	7201.0	4912.1	85.6
30	3	4	5	4348.99	90.3	7200.0	92.8	7200.0	4348.99	4348.99	100.0	6138.5	334.7	82.3
35	3	3	1	**4473.28**	100.0	2808.1	95.5	7200.0	4480.41	4385.81	97.9	7200.4	2684.0	79.7
35	3	3	2	**4438.80**	100.0	1903.8	97.2	7200.0	4637.46	4334.65	93.5	7202.9	6501.0	83.4
35	3	3	3	**5249.48**	98.8	7200.0	97.6	7200.0	-	5085.47	-	7200.38	7061.6	-
35	3	3	4	**4678.54**	88.1	7200.0	92.8	7200.0	5689.84	4427.38	77.8	7206.84	7024.0	86.1
35	3	3	5	**4326.68**	95.1	7200.0	97.5	7200.0	5766.29	4268.16	74.0	7205.17	7002.6	85.5
35	3	4	1	**4980.59**	87.7	7200.0	92.4	7200.0	5026.70	4841.08	96.3	7200.22	3398.9	81.9
35	3	4	2	**4872.03**	95.9	7200.0	94.5	7200.0	5291.20	4757.27	89.9	7210.38	7011.6	85.4
35	3	4	3	6029.70	86.9	7200.0	89.4	7200.0	**5939.48**	5687.89	95.8	7200.26	6653.0	81.3
35	3	4	4	5169.84	80.4	7200.0	90.4	7200.0	**5102.75**	4936.52	96.7	7200.25	4619.8	84.5
35	3	4	5	4958.01	83.5	7200.0	91.1	7200.0	**4873.60**	4763.18	97.7	7203.13	7067.2	82.8
40	3	3	1	**5009.71**	93.7	7200.0	92.7	7200.0	-	4974.97	-	7206.97	7010.0	-
40	3	3	2	**4742.39**	100.0	6590.6	100.0	7200.0	6303.64	4574.25	72.6	7201.65	6590.4	86.7
40	3	3	3	**4747.71**	96.4	7200.0	97.6	7200.0	6071.46	4642.03	76.5	7208.33	7003.0	82.1
40	3	3	4	**4580.04**	96.2	7200.0	93.9	7200.0	4743.90	4423.22	93.2	7208.7	6722.4	81.6
40	3	3	5	**4612.63**	99.1	7200.0	95.2	7200.0	5760.28	4558.94	79.1	7205.2	7007.2	81.7
40	3	4	1	6025.70	74.8	7200.0	83.4	7200.0	**5916.45**	5504.21	93.0	7200.4	6890.5	81.9
40	3	4	2	**5622.76**	81.4	7200.0	86.5	7200.0	7858.53	5156.13	65.6	7207.7	6974.5	89.3
40	3	4	3	**5136.06**	90.0	7200.0	92.9	7200.0	6013.13	4984.39	82.9	7209.1	7002.6	81.8
40	3	4	4	**4855.93**	85.9	7200.0	95.2	7200.0	4941.75	4787.40	96.9	7202.7	6664.8	82.3
40	3	4	5	**5213.30**	87.3	7200.0	90.5	7200.0	5355.59	5051.54	94.3	7208.8	6940.8	80.3
45	3	3	1	**5039.51**	n/a	n/a	n/a	n/a	-	4885.92	-	7204.8	7004.1	-
45	3	3	2	**5296.04**	n/a	n/a	n/a	n/a	-	5155.97	-	7204.8	7004.1	-
45	3	3	3	**4754.62**	n/a	n/a	n/a	n/a	5360.59	4622.71	86.2	7207.0	6913.1	77.3
45	3	3	4	**5458.59**	n/a	n/a	n/a	n/a	7922.57	5229.54	66.0	7204.6	5352.3	86.4
45	3	3	5	**4374.87**	n/a	n/a	n/a	n/a	7280.24	4305.89	59.1	7202.2	7011.2	85.1
45	3	4	1	**5699.41**	n/a	n/a	n/a	n/a	-	5414.06	-	7202.8	7074.1	-
45	3	4	2	**6618.06**	n/a	n/a	n/a	n/a	-	5801.74	-	7204.8	7004.1	-
45	3	4	3	**5183.57**	n/a	n/a	n/a	n/a	6436.43	4952.71	76.9	7206.1	6477.4	81.1
45	3	4	4	**6092.59**	n/a	n/a	n/a	n/a	7451.44	5988.91	80.4	7202.1	7001.7	85.6
45	3	4	5	**4649.79**	n/a	n/a	n/a	n/a	5211.14	4594.77	88.2	7204.8	7001.1	79.0
50	3	3	1	**5827.17**	n/a	n/a	n/a	n/a	7669.07	5570.50	72.6	7207.4	7008.9	84.7
50	3	3	2	**6115.28**	n/a	n/a	n/a	n/a	-	5811.92	-	7201.3	4248.5	-
50	3	3	3	**5590.78**	n/a	n/a	n/a	n/a	-	5291.11	-	7201.3	6248.5	-
50	3	3	4	**5703.97**	n/a	n/a	n/a	n/a	-	5589.50	-	7209.0	6094.6	-
50	3	3	5	**6052.93**	n/a	n/a	n/a	n/a	7724.84	5458.56	70.7	7207.2	6047.5	83.5
50	3	4	1	**7267.52**	n/a	n/a	n/a	n/a	-	6317.92	-	7205.3	6609.7	-
50	3	4	2	**7008.96**	n/a	n/a	n/a	n/a	-	6449.77	-	7200.2	6613.5	-
50	3	4	3	**6566.59**	n/a	n/a	n/a	n/a	7857.83	5792.32	73.7	7203.1	6042.9	84.6
50	3	4	4	**6748.76**	n/a	n/a	n/a	n/a	-	6388.25	-	7202.2	6455.5	-
50	3	4	5	**6764.57**	n/a	n/a	n/a	n/a	8506.97	6152.39	72.3	7201.9	6963.8	85.0

Table 8 Results on multi-vehicle VMIRP instances with three periods and high inventory costs

					Adulyasak et al. (2014)				EXM					
					Veh-Ind		Non-Veh-Ind							
n	l	m	id	Best ub	$\%lb$	t	$\%lb$	t	ub	lb	$\%lb$	t	t_S	$\%rc$
5	3	2	1	2266.61	100.0	0.1	100.0	0.1	2266.61	2266.61	100.0	0.8	0.0	57.4
5	3	2	2	2229.38	100.0	0.0	100.0	0.0	2229.38	2229.38	100.0	0.7	0.0	60.5
5	3	2	3	3698.48	100.0	0.1	100.0	0.2	3698.48	3698.48	100.0	0.9	0.0	63.0
5	3	2	4	2302.44	100.0	0.1	100.0	0.0	2302.44	2302.44	100.0	0.8	0.0	71.8
5	3	2	5	2413.72	100.0	0.0	100.0	0.0	2413.72	2413.72	100.0	0.8	0.0	45.2
5	3	3	1	2414.03	100.0	0.1	100.0	0.1	2414.03	2414.03	100.0	0.6	0.0	59.9
5	3	3	2	2511.05	100.0	0.1	100.0	0.1	2511.05	2511.05	100.0	0.6	0.0	65.0
5	3	3	3	4753.03	100.0	0.0	100.0	0.1	4753.03	4753.03	100.0	0.8	0.0	70.8
5	3	3	4	3132.21	100.0	0.0	100.0	0.1	3132.21	3132.21	100.0	0.7	0.0	79.5
5	3	3	5	2875.58	100.0	0.0	100.0	0.1	2875.58	2875.58	100.0	0.7	0.0	53.8
10	3	2	1	5263.22	100.0	1.4	100.0	0.9	5263.22	5263.22	100.0	6.1	2.9	41.1
10	3	2	2	5464.48	100.0	2.9	100.0	3.2	5464.48	5464.48	100.0	9.2	5.2	53.2
10	3	2	3	4630.08	100.0	1.1	100.0	1.2	4630.08	4630.08	100.0	4.7	1.3	47.6
10	3	2	4	5031.00	100.0	3.2	100.0	7.2	5031.00	5031.00	100.0	5.6	2.2	52.3
10	3	2	5	5318.75	100.0	1.7	100.0	2.5	5318.75	5318.75	100.0	6.1	2.6	41.0
10	3	3	1	5714.31	100.0	3.7	100.0	5.7	5714.31	5714.31	100.0	2.6	0.3	45.6
10	3	3	2	5938.08	100.0	7.3	100.0	3.4	5938.08	5938.08	100.0	3.2	0.8	57.0
10	3	3	3	4919.04	100.0	1.8	100.0	1.3	4919.04	4919.04	100.0	2.5	0.3	50.6
10	3	3	4	5482.86	100.0	9.3	100.0	15.5	5482.86	5482.86	100.0	4.1	1.7	56.1
10	3	3	5	5539.77	100.0	2.6	100.0	2.8	5539.77	5539.77	100.0	2.9	0.7	43.1
15	3	2	1	6108.84	100.0	7.4	100.0	38.9	6108.84	6108.84	100.0	21.3	2.8	36.8
15	3	2	2	6211.69	100.0	4.6	100.0	51.4	6211.69	6211.69	100.0	28.3	6.2	40.9
15	3	2	3	6946.05	100.0	10.9	100.0	8.7	6946.05	6946.05	100.0	21.1	2.2	38.2
15	3	2	4	5551.36	100.0	4.2	100.0	44.0	5551.36	5551.36	100.0	22.8	5.4	44.0
15	3	2	5	5623.74	100.0	4.6	100.0	44.7	5623.74	5623.74	100.0	29.3	7.8	45.4
15	3	3	1	6375.95	100.0	11.6	100.0	45.2	6375.95	6375.95	100.0	12.9	3.8	40.0
15	3	3	2	6533.72	100.0	17.5	100.0	23.5	6533.72	6533.72	100.0	26.9	11.7	43.6
15	3	3	3	7265.38	100.0	14.0	100.0	32.7	7265.38	7265.38	100.0	12.7	3.1	40.9
15	3	3	4	5776.88	100.0	6.7	100.0	6.9	5776.88	5776.88	100.0	14.7	3.4	45.9
15	3	3	5	6068.46	100.0	32.7	100.0	87.3	6068.46	6068.46	100.0	22.7	9.1	49.3
20	3	2	1	7958.82	100.0	130.1	100.0	407.3	7958.82	7958.82	100.0	307.7	90.9	36.4
20	3	2	2	7502.09	100.0	19.5	100.0	21.5	7502.09	7502.09	100.0	73.5	13.9	32.2
20	3	2	3	8228.35	100.0	46.0	100.0	504.1	8228.35	8228.35	100.0	120.1	32.3	34.8
20	3	2	4	7650.75	100.0	60.2	100.0	6873.2	7650.75	7650.75	100.0	248.1	80.6	44.6
20	3	2	5	9027.46	100.0	59.1	100.0	1024.2	9027.46	9027.46	100.0	1588.0	274.8	37.6
20	3	3	1	8398.63	100.0	603.0	100.0	2058.1	8398.63	8398.63	100.0	1301.8	95.8	39.3
20	3	3	2	7641.45	100.0	138.8	100.0	128.4	7641.45	7641.45	100.0	40.2	9.5	33.4
20	3	3	3	8624.22	100.0	407.4	100.0	7200.0	8624.22	8624.22	100.0	331.4	65.0	37.9
20	3	3	4	8270.08	100.0	811.2	96.7	7200.0	8270.08	8270.08	100.0	1508.7	179.3	48.6
20	3	3	5	9656.83	100.0	892.3	100.0	3909.0	9656.83	9656.83	100.0	4731.1	240.4	41.6
25	3	2	1	8923.37	100.0	55.3	100.0	80.5	8923.37	8923.37	100.0	1010.3	553.3	33.4
25	3	2	2	9674.07	100.0	319.0	100.0	7152.1	9674.07	9674.07	100.0	2701.5	959.4	33.8
25	3	2	3	**10368.83**	100.0	160.4	100.0	781.7	11104.06	10206.02	91.9	7200.8	7091.7	36.7
25	3	2	4	8908.31	100.0	33.4	100.0	63.1	8908.31	8908.31	100.0	940.6	708.9	32.7
25	3	2	5	11037.88	100.0	44.2	100.0	26.5	11037.88	11037.88	100.0	4371.6	3786.4	27.8
25	3	3	1	9277.31	100.0	234.9	100.0	1954.7	9277.31	9277.31	100.0	3860.0	286.3	36.1
25	3	3	2	10048.88	100.0	1753.3	98.7	7200.0	10048.88	10043.67	99.9	7200.2	542.8	36.4
25	3	3	3	11022.72	100.0	1857.7	99.1	7200.0	11022.72	10982.11	99.6	7200.2	385.4	35.8
25	3	3	4	9334.96	100.0	233.0	99.1	7200.0	9334.96	9334.96	100.0	1149.9	319.5	35.7
25	3	3	5	11715.70	100.0	1527.6	100.0	3195.1	11715.70	11549.05	98.6	7200.2	3420.8	32.1
30	3	3	1	**13390.84**	100.0	2296.7	98.9	7200.0	14392.44	13320.81	92.6	7201.3	7050.4	33.2
30	3	3	2	11911.63	100.0	435.5	100.0	5334.6	11911.63	11835.94	99.4	7200.2	806.3	29.0
30	3	3	3	13003.14	100.0	1524.9	99.1	7200.0	13003.14	12995.29	99.9	7200.2	194.0	25.4
30	3	3	4	10619.39	100.0	4296.7	98.0	7200.0	10619.39	10527.89	99.1	7200.2	1500.9	33.2
30	3	3	5	10911.59	100.0	6708.9	97.6	7200.0	10911.59	10884.44	99.8	7200.2	796.1	29.5
30	3	4	1	13976.21	94.8	7200.0	97.3	7200.0	13976.21	13830.47	99.0	7200.2	1396.7	31.0
30	3	4	2	12252.36	100.0	2110.7	99.0	7200.0	12252.36	12224.56	99.8	7200.2	447.9	30.9
30	3	4	3	13324.90	97.6	7200.0	98.8	7200.0	13324.90	13293.43	99.8	7200.2	317.3	26.9
30	3	4	4	**11027.66**	95.5	7200.0	97.4	7200.0	11049.15	10931.71	98.9	7200.2	3393.2	36.4
30	3	4	5	11270.83	95.8	7200.0	97.1	7200.0	**11270.83**	11270.83	100.0	6844.6	283.1	31.9
35	3	3	1	**12764.43**	99.7	7200.0	98.4	7200.0	13530.55	12694.97	93.8	7202.0	6962.7	32.1
35	3	3	2	**11352.68**	100.0	817.2	98.8	7200.0	11449.35	11211.09	97.9	7203.2	6970.3	32.9
35	3	3	3	**15278.37**	100.0	5074.4	99.4	7200.0	15566.09	15149.00	97.3	7200.3	6619.8	28.6
35	3	3	4	**11746.39**	96.9	7200.0	97.1	7200.0	11873.09	11399.20	96.0	7205.1	7009.2	33.8
35	3	3	5	**11881.10**	98.5	7200.0	99.3	7200.0	11946.74	11821.76	99.0	7200.3	2379.2	29.9
35	3	4	1	13426.45	93.8	7200.0	96.0	7200.0	**13296.40**	13140.71	98.8	7200.2	1841.2	30.9
35	3	4	2	**11786.79**	99.0	7200.0	97.8	7200.0	12352.07	11560.36	93.6	7204.9	7007.2	38.0
35	3	4	3	**16151.59**	94.8	7200.0	95.7	7200.0	16696.15	15725.52	94.2	7227.3	7016.8	33.1
35	3	4	4	12295.65	91.7	7200.0	95.5	7200.0	**12160.17**	11944.50	98.2	7200.2	2744.1	35.4
35	3	4	5	12701.76	92.7	7200.0	94.9	7200.0	**12448.27**	12263.81	98.5	7203.0	7010.6	32.7
40	3	3	1	**14733.71**	97.4	7200.0	97.9	7200.0	-	14707.52	-	7207.6	6560.4	-
40	3	3	2	**12267.02**	100.0	6515.8	100.0	4039.2	13699.71	11987.27	87.5	7202.7	6724.4	39.0
40	3	3	3	**14525.45**	99.5	7200.0	99.1	7200.0	15289.70	14235.11	93.1	7206.1	6897.4	29.0
40	3	3	4	**12401.25**	99.3	7200.0	98.0	7200.0	12571.11	12178.95	96.9	7206.0	7044.1	31.1
40	3	3	5	**14143.72**	100.0	5384.6	98.5	7200.0	16256.25	14121.72	86.9	7208.1	7002.3	35.1
40	3	4	1	16057.77	89.7	7200.0	91.7	7200.0	**15701.07**	15232.85	97.0	7202.3	7017.6	31.2
40	3	4	2	13022.49	93.0	7200.0	95.6	7200.0	**12940.30**	12568.74	97.1	7202.8	6929.8	35.4
40	3	4	3	**14898.49**	96.3	7200.0	98.0	7200.0	15005.16	14677.13	97.8	7200.2	1786.5	27.7
40	3	4	4	**12725.05**	93.8	7200.0	97.7	7200.0	12736.43	12571.78	98.7	7200.6	3602.3	32.1
40	3	4	5	14763.54	95.6	7200.0	96.3	7200.0	**14721.19**	14504.47	98.5	7200.5	6002.3	28.1
45	3	3	1	**15327.90**	n/a	n/a	n/a	n/a	17526.21	15259.30	87.1	7200.5	6909.8	34.9
45	3	3	2	**14818.70**	n/a	n/a	n/a	n/a	-	14679.11	-	7202.3	6824.8	-
45	3	3	3	**15744.72**	n/a	n/a	n/a	n/a	16425.69	15476.52	94.2	7205.7	6387.6	25.9
45	3	3	4	**15015.64**	n/a	n/a	n/a	n/a	15811.58	14598.18	92.3	7208.1	6918.2	32.8
45	3	3	5	**14305.11**	n/a	n/a	n/a	n/a	16490.40	13981.35	84.8	7200.6	6227.5	33.4
45	3	4	1	**15997.64**	n/a	n/a	n/a	n/a	17327.67	15448.13	89.2	7207.4	6815.7	34.0
45	3	4	2	**16025.69**	n/a	n/a	n/a	n/a	-	15293.57	-	7205.5	6943.3	-
45	3	4	3	**16252.17**	n/a	n/a	n/a	n/a	16267.81	15875.39	97.6	7200.2	3264.7	24.9
45	3	4	4	15857.40	n/a	n/a	n/a	n/a	**15601.74**	15209.43	97.5	7200.2	5890.2	31.9
45	3	4	5	**14720.37**	n/a	n/a	n/a	n/a	15591.13	14379.41	92.2	7201.6	6765.3	29.3
50	3	3	1	**16518.46**	n/a	n/a	n/a	n/a	18176.81	16386.24	90.1	7206.6	6546.1	35.4
50	3	3	2	**16571.41**	n/a	n/a	n/a	n/a	-	16146.52	-	7206.3	4739.8	-
50	3	3	3	**16467.13**	n/a	n/a	n/a	n/a	18804.66	15857.25	84.3	7206.6	7026.1	35.9
50	3	3	4	**17867.88**	n/a	n/a	n/a	n/a	-	17747.74	-	7202.9	5688.6	-
50	3	3	5	**17174.43**	n/a	n/a	n/a	n/a	18942.46	16764.88	88.5	7209.7	6645.4	32.9
50	3	4	1	**17485.21**	n/a	n/a	n/a	n/a	-	16692.66	-	7203.7	7064.6	-
50	3	4	2	**18045.92**	n/a	n/a	n/a	n/a	18147.11	16737.64	92.2	7208.1	7093.7	35.8
50	3	4	3	**17325.20**	n/a	n/a	n/a	n/a	18871.26	16486.18	87.4	7208.2	5915.2	36.2
50	3	4	4	**19139.80**	n/a	n/a	n/a	n/a	-	18494.81	-	7201.1	7026.8	-
50	3	4	5	18405.06	n/a	n/a	n/a	n/a	**18303.42**	17360.96	94.9	7202.2	6051.2	30.5

Table 9 Results on multi-vehicle VMIRP instances with six periods and low inventory costs

				Adulyasak et al. (2014)					EXM					
					Veh-Ind		Non-Veh-Ind							
n	l	m	id	Best ub	$\%lb$	t	$\%lb$	t	ub	lb	$\%lb$	t	t_S	$\%rc$
5	6	2	1	4286.37	100.0	2.2	100.0	7.0	4286.37	4286.37	100.0	1.5	0.3	93.4
5	6	2	2	3869.33	100.0	3.1	100.0	12.9	3869.33	3869.33	100.0	2.0	0.5	93.2
5	6	2	3	5997.42	100.0	0.6	100.0	1.2	5997.42	5997.42	100.0	1.0	0.1	96.0
5	6	2	4	3814.27	100.0	1.2	100.0	1.3	3814.27	3814.27	100.0	1.2	0.2	94.5
5	6	2	5	3062.01	100.0	0.5	100.0	0.8	3062.01	3062.01	100.0	1.3	0.2	92.0
5	6	3	1	5258.03	100.0	0.3	100.0	1.5	5258.03	5258.03	100.0	0.9	0.0	94.6
5	6	3	2	4881.51	100.0	0.1	100.0	0.1	4881.51	4881.51	100.0	0.9	0.1	94.7
5	6	3	3	10555.98	100.0	0.0	100.0	1.5	10555.98	10555.98	100.0	1.4	0.1	97.7
5	6	3	4	5028.36	100.0	0.3	100.0	14.2	5028.36	5028.36	100.0	1.0	0.1	95.8
5	6	3	5	4392.47	100.0	0.2	100.0	0.9	4392.47	4392.47	100.0	1.3	0.0	94.3
10	6	2	1	5923.33	100.0	181.3	96.4	7200.0	5923.33	5923.33	100.0	177.7	11.4	91.9
10	6	2	2	6927.65	100.0	621.0	100.0	598.2	6927.65	6927.65	100.0	2199.4	10.2	94.6
10	6	2	3	5533.68	100.0	118.2	100.0	723.2	5533.68	5533.68	100.0	511.2	12.0	92.2
10	6	2	4	6463.46	100.0	79.7	100.0	2585.2	6463.46	6463.46	100.0	293.4	16.2	93.5
10	6	2	5	5441.76	100.0	97.8	100.0	371.9	5441.76	5441.76	100.0	1391.5	6.0	89.8
10	6	3	1	7535.66	95.8	7200.0	91.5	7200.0	7535.66	7535.66	100.0	176.1	11.9	93.6
10	6	3	2	8594.37	100.0	6709.7	100.0	7200.0	8594.37	8594.37	100.0	290.0	15.4	95.7
10	6	3	3	6604.50	100.0	655.8	100.0	2816.0	6604.50	6604.50	100.0	829.1	9.4	93.5
10	6	3	4	7891.84	100.0	2387.7	95.7	7200.0	7891.84	7891.84	100.0	1031.9	15.1	94.7
10	6	3	5	6253.99	100.0	1571.1	100.0	2828.9	6253.99	6253.99	100.0	260.5	8.1	91.2
15	6	2	1	**6143.34**	100.0	654.2	96.1	7200.0	6165.38	6001.02	97.3	7200.3	33.7	88.1
15	6	2	2	**6382.11**	100.0	203.4	98.9	7200.0	6383.11	6222.30	97.5	7200.2	69.1	88.8
15	6	2	3	7239.38	100.0	1062.1	95.2	7200.0	7239.38	7095.81	98.0	7200.4	48.4	88.6
15	6	2	4	**6465.37**	100.0	720.6	97.0	7200.0	6494.35	6260.01	96.4	7200.2	92.3	91.2
15	6	2	5	**6669.99**	100.0	1040.3	94.7	7200.0	6875.90	6413.55	93.3	7200.2	137.1	91.7
15	6	3	1	**7137.36**	92.9	7200.0	91.9	7200.0	7304.36	6942.96	95.1	7200.2	99.8	90.0
15	6	3	2	**7559.90**	95.7	7200.0	93.1	7200.0	7649.50	7292.03	95.3	7200.2	104.9	90.6
15	6	3	3	**8465.21**	92.8	7200.0	91.4	7200.0	8539.91	8310.14	97.3	7200.2	104.4	90.3
15	6	3	4	**7604.86**	94.0	7200.0	92.8	7200.0	7618.82	7352.53	96.5	7200.2	84.6	92.5
15	6	3	5	**8046.60**	91.9	7200.0	92.4	7200.0	8224.07	7681.01	93.4	7200.2	218.1	93.1
20	6	2	1	**7704.14**	n/a	n/a	n/a	n/a	8086.34	7330.65	90.7	7200.2	2426.7	88.8
20	6	2	2	**6569.33**	n/a	n/a	n/a	n/a	6675.70	6379.43	95.6	7200.3	452.5	85.6
20	6	2	3	**7781.52**	n/a	n/a	n/a	n/a	8013.64	7532.06	94.0	7200.2	401.2	89.4
20	6	2	4	**8746.71**	n/a	n/a	n/a	n/a	9128.25	8294.62	90.9	7200.3	1596.3	91.4
20	6	2	5	**8968.73**	n/a	n/a	n/a	n/a	9377.61	8479.43	90.4	7204.4	1681.6	89.7
20	6	3	1	**9690.78**	n/a	n/a	n/a	n/a	9860.50	8842.56	89.7	7200.2	1181.9	90.8
20	6	3	2	**7477.33**	n/a	n/a	n/a	n/a	7527.54	7040.63	93.5	7200.2	236.8	87.2
20	6	3	3	**9335.74**	n/a	n/a	n/a	n/a	9080.12	8574.27	94.4	7200.2	181.9	90.6
20	6	3	4	10989.06	n/a	n/a	n/a	n/a	**10846.92**	10028.15	92.5	7200.2	433.2	92.7
20	6	3	5	11826.34	n/a	n/a	n/a	n/a	**11357.43**	10557.12	93.0	7200.2	4056.8	91.5
25	6	2	1	**7892.30**	n/a	n/a	n/a	n/a	8657.11	7573.02	87.5	7200.2	6616.6	89.2
25	6	2	2	**8667.63**	n/a	n/a	n/a	n/a	9780.01	8260.99	84.5	7202.8	6873.2	89.5
25	6	2	3	**8950.41**	n/a	n/a	n/a	n/a	9754.76	8739.19	89.6	7200.2	6788.5	88.1
25	6	2	4	**7954.64**	n/a	n/a	n/a	n/a	8549.26	7814.72	91.4	7202.0	6990.8	88.5
25	6	2	5	**9049.47**	n/a	n/a	n/a	n/a	9374.68	8587.22	91.6	7202.8	6451.6	86.3
25	6	3	1	**9024.73**	n/a	n/a	n/a	n/a	9375.30	8420.46	89.8	7200.2	1541.7	90.0
25	6	3	2	11136.70	n/a	n/a	n/a	n/a	**10430.03**	9662.85	92.6	7200.3	3034.9	90.1
25	6	3	3	**11320.78**	n/a	n/a	n/a	n/a	14146.16	10398.27	73.5	7200.3	7000.5	91.8
25	6	3	4	**9233.14**	n/a	n/a	n/a	n/a	9338.73	8760.84	93.8	7200.3	1917.9	89.5
25	6	3	5	11760.03	n/a	n/a	n/a	n/a	**11037.56**	10346.64	93.7	7200.3	1671.5	88.4
30	6	3	1	n/a	n/a	n/a	n/a	n/a	14115.87	10493.40	74.3	7202.1	6910.6	88.4
30	6	3	2	n/a	n/a	n/a	n/a	n/a	10323.19	9334.39	90.4	7202.3	6924.4	86.7
30	6	3	3	n/a	n/a	n/a	n/a	n/a	10449.54	9428.62	90.2	7208.8	5297.7	83.8
30	6	3	4	n/a	n/a	n/a	n/a	n/a	10719.44	9183.27	85.7	7200.2	6954.6	89.5
30	6	3	5	n/a	n/a	n/a	n/a	n/a	10071.78	9058.20	89.9	7203.5	6262.8	87.1
30	6	4	1	n/a	n/a	n/a	n/a	n/a	13066.99	12092.09	92.5	7200.3	3482.9	87.4
30	6	4	2	n/a	n/a	n/a	n/a	n/a	11914.42	10647.63	89.4	7200.3	2290.7	88.5
30	6	4	3	n/a	n/a	n/a	n/a	n/a	11269.56	10341.72	91.8	7200.2	1508.8	85.0
30	6	4	4	n/a	n/a	n/a	n/a	n/a	11881.20	10446.30	87.9	7200.7	6699.0	90.5
30	6	4	5	n/a	n/a	n/a	n/a	n/a	11407.76	10232.09	89.7	7200.2	2292.0	88.6

Table 10 Results on multi-vehicle VMIRP instances with six periods and high inventory costs

				Adulyasak et al. (2014)					EXM					
					Veh-Ind		Non-Veh-Ind							
n	l	m	id	Best ub	$\%lb$	t	$\%lb$	t	ub	lb	$\%lb$	t	t_S	$\%rc$
5	6	2	1	6888.73	100.0	2.0	100.0	3.5	6888.73	6888.73	100.0	1.7	0.3	58.1
5	6	2	2	6202.65	100.0	2.4	100.0	15.5	6202.65	6202.65	100.0	2.3	0.5	58.3
5	6	2	3	8183.76	100.0	1.0	100.0	1.2	8183.76	8183.76	100.0	1.0	0.1	70.3
5	6	2	4	5726.62	100.0	1.1	100.0	1.3	5726.62	5726.62	100.0	1.1	0.1	62.9
5	6	2	5	5234.05	100.0	0.5	100.0	0.6	5234.05	5234.05	100.0	1.1	0.1	53.8
5	6	3	1	7864.79	100.0	0.2	100.0	0.8	7864.79	7864.79	100.0	0.8	0.0	63.2
5	6	3	2	7174.59	100.0	0.1	100.0	0.1	7174.59	7174.59	100.0	0.8	0.1	64.4
5	6	3	3	12780.57	100.0	0.0	100.0	0.3	12780.57	12780.57	100.0	0.8	0.1	80.7
5	6	3	4	6933.86	100.0	0.3	100.0	18.0	6933.86	6933.86	100.0	0.9	0.1	69.5
5	6	3	5	6593.51	100.0	0.2	100.0	0.9	6593.51	6593.51	100.0	0.7	0.0	62.8
10	6	2	1	10294.48	100.0	240.2	99.2	7200.0	10294.48	10294.48	100.0	177.2	6.2	52.9
10	6	2	2	10265.25	100.0	785.2	100.0	1944.9	10265.25	10265.25	100.0	2182.1	27.5	63.9
10	6	2	3	9400.47	100.0	102.9	100.0	669.0	9400.47	9400.47	100.0	420.6	9.8	54.3
10	6	2	4	10161.40	100.0	150.7	100.0	1986.1	10161.40	10161.40	100.0	302.2	10.8	59.5
10	6	2	5	10391.07	100.0	82.9	100.0	341.9	10391.07	10391.07	100.0	316.4	6.5	47.1
10	6	3	1	11900.94	96.4	7200.0	94.4	7200.0	11900.94	11900.94	100.0	111.8	9.5	59.3
10	6	3	2	11929.85	100.0	5313.3	99.4	7200.0	11929.85	11929.85	100.0	133.6	11.3	68.9
10	6	3	3	10456.78	100.0	871.2	100.0	2929.3	10456.78	10456.78	100.0	1258.2	10.2	59.0
10	6	3	4	11598.75	100.0	2521.3	95.6	7200.0	11598.75	11598.75	100.0	1354.5	12.9	64.4
10	6	3	5	11192.68	100.0	1903.0	100.0	1962.0	11192.68	11192.68	100.0	268.6	7.2	51.0
15	6	2	1	**12825.51**	100.0	246.4	97.7	7200.0	12830.99	12648.57	98.6	7200.2	41.0	42.4
15	6	2	2	**12821.22**	100.0	263.1	99.4	7200.0	12826.37	12623.23	98.4	7200.3	58.3	44.3
15	6	2	3	**14733.15**	100.0	1251.9	97.6	7200.0	14796.15	14582.63	98.6	7200.3	44.7	43.8
15	6	2	4	**11592.15**	100.0	555.8	98.1	7200.0	11605.05	11442.76	98.6	7200.2	92.4	50.9
15	6	2	5	11748.70	100.0	1498.5	97.8	7200.0	11748.70	11504.56	97.9	7200.2	118.3	51.9
15	6	3	1	13785.72	96.5	7200.0	95.6	7200.0	13785.72	13624.12	98.8	7200.2	82.5	46.5
15	6	3	2	**14009.39**	97.0	7200.0	96.3	7200.0	14095.52	13693.86	97.2	7200.2	113.0	49.2
15	6	3	3	**15974.42**	95.5	7200.0	95.2	7200.0	16012.42	15791.81	98.6	7200.2	72.9	47.9
15	6	3	4	**12734.73**	97.4	7200.0	96.1	7200.0	12887.45	12449.05	96.6	7200.3	141.1	55.8
15	6	3	5	**13134.22**	95.5	7200.0	94.9	7200.0	13232.49	12715.72	96.1	7200.3	155.7	57.5
20	6	2	1	**15946.57**	n/a	n/a	n/a	n/a	16098.83	15393.92	95.6	7200.3	2600.4	43.3
20	6	2	2	**15144.35**	n/a	n/a	n/a	n/a	15191.83	14882.20	98.0	7200.3	260.8	37.4
20	6	2	3	**15366.82**	n/a	n/a	n/a	n/a	15503.94	14968.21	96.5	7200.3	442.9	45.6
20	6	2	4	**15850.97**	n/a	n/a	n/a	n/a	16274.16	15242.53	93.7	7200.3	1118.5	51.5
20	6	2	5	**17654.71**	n/a	n/a	n/a	n/a	17909.43	16972.18	94.8	7200.3	1527.5	46.1
20	6	3	1	**17853.11**	n/a	n/a	n/a	n/a	17932.95	16974.28	94.7	7200.2	821.3	48.9
20	6	3	2	**15939.81**	n/a	n/a	n/a	n/a	15982.33	15498.24	97.0	7200.3	232.5	40.4
20	6	3	3	16871.90	n/a	n/a	n/a	n/a	**16825.56**	16027.68	95.3	7200.3	248.2	49.9
20	6	3	4	18130.74	n/a	n/a	n/a	n/a	**17834.74**	16987.27	95.2	7200.2	581.4	55.7
20	6	3	5	20298.58	n/a	n/a	n/a	n/a	20429.06	19003.73	93.0	7200.3	835.4	52.8
25	6	2	1	**16383.06**	n/a	n/a	n/a	n/a	16915.50	15868.56	93.8	7200.7	6402.9	44.4
25	6	2	2	**18060.15**	n/a	n/a	n/a	n/a	20127.08	17275.64	85.8	7205.1	7026.8	48.6
25	6	2	3	**19352.31**	n/a	n/a	n/a	n/a	20406.40	18873.77	92.5	7201.1	7097.1	43.5
25	6	2	4	**16781.57**	n/a	n/a	n/a	n/a	17181.65	16388.07	95.4	7205.9	6825.6	43.1
25	6	2	5	**20682.66**	n/a	n/a	n/a	n/a	22370.02	19923.07	89.1	7205.5	6941.5	42.5
25	6	3	1	**17426.30**	n/a	n/a	n/a	n/a	17832.54	16705.73	93.7	7200.3	1290.5	47.1
25	6	3	2	20477.42	n/a	n/a	n/a	n/a	**19624.32**	18790.26	95.7	7200.3	2616.6	47.2
25	6	3	3	**21748.29**	n/a	n/a	n/a	n/a	21910.48	20570.19	93.9	7200.3	1630.2	47.4
25	6	3	4	**18001.88**	n/a	n/a	n/a	n/a	18240.43	17357.35	95.2	7200.3	2856.9	46.3
25	6	3	5	23562.27	n/a	n/a	n/a	n/a	-	21688.16	-	7203.9	7115.1	-
30	6	3	1	n/a	n/a	n/a	n/a	n/a	27412.68	25034.46	91.3	7204.6	6913.0	39.9
30	6	3	2	n/a	n/a	n/a	n/a	n/a	22580.76	21445.55	95.0	7208.8	3137.7	39.4
30	6	3	3	n/a	n/a	n/a	n/a	n/a	25523.43	24282.77	95.1	7200.3	1979.3	34.1
30	6	3	4	n/a	n/a	n/a	n/a	n/a	20686.07	18930.76	91.5	7205.4	6402.6	45.9
30	6	3	5	n/a	n/a	n/a	n/a	n/a	21665.65	20412.67	94.2	7206.8	4814.3	40.4
30	6	4	1	n/a	n/a	n/a	n/a	n/a	28197.79	26702.31	94.7	7200.2	1828.9	41.5
30	6	4	2	n/a	n/a	n/a	n/a	n/a	24124.71	22720.94	94.2	7200.3	4235.9	43.2
30	6	4	3	n/a	n/a	n/a	n/a	n/a	26545.07	25212.25	95.0	7200.3	1282.1	36.7
30	6	4	4	n/a	n/a	n/a	n/a	n/a	21976.70	20205.37	91.9	7200.2	4461.9	49.1
30	6	4	5	n/a	n/a	n/a	n/a	n/a	22854.80	21564.19	94.4	7200.2	1673.7	43.6

References

Adulyasak, Y., J.-F. Cordeau, R. Jans. 2014a. Formulations and branch-and-cut algorithms for multivehicle production and inventory routing problems. *INFORMS Journal on Computing* **26**(1) 103–120.

Adulyasak, Y., J.-F. Cordeau, R. Jans. 2014b. Optimization-based adaptive large neighborhood search for the production routing problem. *Transportation Science* **48**(1) 20–45.

Adulyasak, Y., J.-F. Cordeau, R. Jans. 2015. The production routing problem: A review of formulations and solution algorithms. *Computers & Operations Research* **55** 141–152.

Andersson, H., A. Hoff, M. Christiansen, G. Hasle, A. Løkketangen. 2010. Industrial aspects and literature survey: Combined inventory management and routing. *Computers & Operations Research* **37**(9) 1515–1536.

Archetti, C., L. Bertazzi, A. Hertz, M. G. Speranza. 2012. A hybrid heuristic for an inventory routing problem. *INFORMS Journal on Computing* **24**(1) 101–116.

Archetti, C., L. Bertazzi, G. Laporte, M. G. Speranza. 2007. A branch-and-cut algorithm for a vendor-managed inventory-routing problem. *Transportation Science* **41**(3) 382–391.

Archetti, C., L. Bertazzi, G. Paletta, M. G. Speranza. 2011. Analysis of the maximum level policy in a production-distribution system. *Computers & Operations Research* **38**(12) 1731–1746.

Avella, P., M. Boccia, L. A. Wolsey. 2015. Single-item reformulations for a vendor managed inventory routing problem: Computational experience with benchmark instances. *Networks* **65**(2) 129–138.

Avella, P., M. Boccia, L. A. Wolsey. 2018. Single-period cutting planes for inventory routing problems. *Transportation Science* **52**(3) 497–508.

Baldacci, R., E. Bartolini, A. Mingozzi, R. Roberti. 2010. An exact solution framework for a broad class of vehicle routing problems. *Computational Management Science* **7** 229–268.

Baldacci, R., N. Christofides, A. Mingozzi. 2008. An exact algorithm for the vehicle routing problem based on the set partitioning formulation with additional cuts. *Mathematical Programming Ser. A* **115**(2) 351–385.

Baldacci, R., A. Mingozzi, R. Roberti. 2011. New route relaxation and pricing strategies for the vehicle routing problem. *Operations Research* **59**(5) 1269–1283.

Bard, Jonathan F, Narameth Nananukul. 2010. A branch-and-price algorithm for an integrated production and inventory routing problem. *Computers & Operations Research* **37**(12) 2202–2217.

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

Bertazzi, L., G. Paletta, M. G. Speranza. 2002. Deterministic order-up-to level policies in an inventory routing problem. *Transportation Science* **36**(1) 119–132.

Bertazzi, L., M. Savelsbergh, M.G. Speranza. 2008. Inventory routing. Bruce Golden, S. Raghavan, Edward Wasil, eds., *Operations Research/Computer Science Interfaces*, vol. 43. Springer, New York, 49–72.

Campbell, Ann M., Lloyd W. Clarke, Martin W. P. Savelsbergh. 2002. Inventory routing in practice. P. Toth, Vigo. D., eds., *The Vehicle Routing Problem*. SIAM, Philadelphia, 309–330.

Codato, G., M. Fischetti. 2006. Combinatorial benders' cuts for mixed-integer linear programming. *Operations Research* **54**(4) 756–766.

Coelho, L. C., J.-F. Cordeau, G. Laporte. 2014. Thirty years of inventory routing. *Transportation Science* **48**(1) 1–19.

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

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

Desaulniers, G., J. G. Rakke, L. C. Coelho. 2016. A branch-price-and-cut algorithm for the inventory-routing problem. *Transportation Science* **50**(3) 1060–1076.

Gendreau, Michel, Alain Hertz, Gilbert Laporte. 1994. A tabu search heuristic for the vehicle routing problem. *Management Science* **40**(10) 1276–1290.

Geoffrion, A. M. 1972. Generalized benders decomposition. *Journal of Optimization Theory and Applications* **10**(4) 237–260.

Hooker, J. N. 2000. *Logic-Based Methods for Optimization*. John Wiley & Sons, Inc.

Hooker, J. N. 2007. Planning and scheduling by logic-based benders decomposition. *Operations Research* **55**(3) 588–602.

Hooker, J. N., G. Ottosson. 2003. Logic-based benders decomposition. *Mathematical Programming* **96**(1) 33–60.

IBM CPLEX. 2016. *IBM ILOG CPLEX 12.6.3 callable library*.

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

Pessoa, A., M. P. de Aragão, E. Uchoa. 2008. Robust branch-cut-and-price algorithms for vehicle routing problems. B.L. Golden, S. Raghavan, E. Wasil, eds., *The Vehicle Routing Problem: Latest Advances and New Challenges*, vol. 43. Springer Verlag, 297–325.

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

Poggi, M., E. Uchoa. 2014. Chapter 3: New exact algorithms for the capacitated vehicle routing problem. *MOS-SIAM Series on Optimization*. Society for Industrial and Applied Mathematics, 59–86.

Solyalı, O., J.-F. Cordeau, G. Laporte. 2012. Robust inventory routing under demand uncertainty. *Transportation Science* **46**(3) 327–340.

Solyalı, O., H. Süral. 2011. A branch-and-cut algorithm using a strong formulation and an a priori tour-based heuristic for an inventory-routing problem. *Transportation Science* **45**(3) 335–345.

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

Zhang, Z., Z. Luo, H. Qin, A. Lim. 2019. Exact algorithms for the vehicle routing problem with time windows and combinatorial auction. *Transportation Science* **53**(2) 427–441.