

This article was downloaded by: ["University at Buffalo Libraries"]

On: 20 April 2013, At: 09:49

Publisher: Taylor & Francis

Informa Ltd Registered in England and Wales Registered Number: 1072954 Registered office: Mortimer House, 37-41 Mortimer Street, London W1T 3JH, UK



International Journal of Production Research

Publication details, including instructions for authors and subscription information:

<http://www.tandfonline.com/loi/tprs20>

A branch-and-cut algorithm for the multi-product multi-vehicle inventory-routing problem

Leandro C. Coelho^a & Gilbert Laporte^a

^a CIRRELT and HEC Montréal, 3000 chemin de la Côte-Sainte-Catherine, Montréal, Canada, H3T 2A7

Version of record first published: 14 Feb 2013.

To cite this article: Leandro C. Coelho & Gilbert Laporte (2013): A branch-and-cut algorithm for the multi-product multi-vehicle inventory-routing problem, International Journal of Production Research, DOI:10.1080/00207543.2012.757668

To link to this article: <http://dx.doi.org/10.1080/00207543.2012.757668>

PLEASE SCROLL DOWN FOR ARTICLE

Full terms and conditions of use: <http://www.tandfonline.com/page/terms-and-conditions>

This article may be used for research, teaching, and private study purposes. Any substantial or systematic reproduction, redistribution, reselling, loan, sub-licensing, systematic supply, or distribution in any form to anyone is expressly forbidden.

The publisher does not give any warranty express or implied or make any representation that the contents will be complete or accurate or up to date. The accuracy of any instructions, formulae, and drug doses should be independently verified with primary sources. The publisher shall not be liable for any loss, actions, claims, proceedings, demand, or costs or damages whatsoever or howsoever caused arising directly or indirectly in connection with or arising out of the use of this material.

A branch-and-cut algorithm for the multi-product multi-vehicle inventory-routing problem

Leandro C. Coelho and Gilbert Laporte*

CIRRELT and HEC Montréal, 3000 chemin de la Côte-Sainte-Catherine, Montréal, Canada H3T 2A7

(Final version received 6 December 2012)

The combined operation of distribution and inventory control achieved through a vendor-managed inventory strategy creates a synergetic interaction that benefits supplier and customers. Inventory-Routing Problems (IRPs) arise when inventory and routing decisions must be taken simultaneously, which yields a difficult combinatorial optimisation problem. While most IRP research deals with a single product, there are often several products involved in distribution activities. In this paper, we propose a branch-and-cut algorithm for the solution of IRPs with multiple products and multiple vehicles. We formally define and model the problem, and we solve it exactly. We also consider the inclusion of consistency features that are meaningful in a multi-product environment and help improve the quality of the service offered.

Keywords: operational research; optimisation; inventory management; vendor managed inventory; inventory-routing; branch-and-cut

1. Introduction

Inventory-Routing Problems (IRPs) have received increased attention in recent years. Several heuristics (Bertazzi, Paletta, and Speranza 2002; Archetti et al. 2012; Coelho, Cordeau, and Laporte 2012a) and exact algorithms (Archetti et al. 2007; Solyali and Süral 2011) are available for its single-vehicle and single-product version. Recently, the multi-vehicle case (MIRP) was also solved heuristically (Coelho, Cordeau, and Laporte 2012b) and exactly (Adulyasak, Cordeau, and Jans 2012; Coelho and Laporte 2013). In all these papers, there is only one product, whereas many vendor-managed inventory (VMI) applications are concerned with the distribution of several products. In this paper we model and optimally solve the multi-product multi-vehicle IRP (MMIRP).

Several applications of the MMIRP are well documented. Most of these arise in maritime logistics, namely in the distribution of several types of fuel and gases by compartmentalised ships (Bausch, Brown, and Ronen 1998; Qu, Bookbinder, and Iyogun 1999; Bertazzi and Speranza 2002; Ronen 2002; Persson and Göthe-Lundgren 2005; Grønhaug et al. 2010; Christiansen et al. 2011; Uggen, Fodstad, and Nørstebø 2011; Engineer et al. 2012; Stålhane et al. 2012). Non-maritime applications include the distribution of perishable products (Federgruen and Zipkin 1984; Federgruen, Prastacos, and Zipkin 1986), the transportation of gases by tanker trucks (Bell et al. 1983), the automobile components industry (Alegre, Laguna, and Pacheco 2007), and fuel delivery (Popović, Vidović, and Radivojević 2012).

Like the applications, the assumptions and solution procedures are also diverse. A Lagrangian-based heuristic based on a VRP algorithm was proposed by Chien, Balakrishnan, and Wong (1989). A special case of the problem with a single customer and predetermined frequency deliveries was studied by Speranza and Ukovich (1994, 1996). Bertazzi, Speranza, and Ukovich (1997) later expanded these studies to handle multiple customers, in which each customer is treated individually, and those with the same optimal frequency are aggregated for the computation of routes. Carter et al. (1996) proposed a two-phase heuristic that first solves an allocation problem to determine when and how much to deliver to customers, and then constructs delivery routes. Sindhuchao et al. (2005) assumes that each vehicle always carries the same set of items and then formulates the IRP as a set partitioning problem. An MMIRP with demand uncertainty was studied by Huang and Lin (2010), who solved it by means of an ant colony optimisation algorithm. A variation of the multi-product version that also considers multiple suppliers but only one customer (many-to-one structure) was analysed by Moin, Salhi, and Aziz (2011). The authors derive lower and upper bounds after solving a linear mathematical formulation with a commercial solver and then compute better upper bounds by means of a genetic algorithm. Building upon the previous structure, Ramkumar et al. (2012) proposed a MILP for a many-to-many multi-item multi-depot IRP. However, computational results showed the limitations of the method since small instances with only

*Corresponding author. Email: gilbert.laporte@cirrelt.ca

two vehicles, two products, two suppliers, three customers and three periods could not be solved to optimality within eight hours of computing time. Despite the relatively large number of papers in this field, no comparison between algorithms is possible due to the different assumptions made in each paper and to the lack of a common test bed.

We consider a version of the MMIRP in which a supplier is responsible for the distribution of several products to a set of geographically dispersed customers using a fleet of vehicles. Customers and vehicles have a maximum inventory capacity that is shared by all products. This problem arises, for example, in the grocery and beverage industries. We deal with a deterministic version of the problem in which the supplier has full knowledge of future demands, such that no stockout at the customers is allowed to occur. We propose an exact algorithm for this problem. We increase the scope of an exact approach based on a branch-and-cut algorithm put forward for the single-vehicle case by Archetti et al. (2007) and extended to the MIRP by Adulyasak, Cordeau, and Jans (2012) and by Coelho and Laporte (2013). Our implementation is shown to obtain optimal or high-quality solutions for medium-sized instances of the problem. Using our model and algorithm we are able to evaluate the impact of each parameter of the problem, namely the number of customers, products, vehicles and periods, in terms of computational difficulty and solution cost. We also evaluate the impact of ensuring stable operations over time through the inclusion of consistency features (Coelho, Cordeau, and Laporte 2012b). We measure the effect of these features in a multi-product context.

The objective of this paper is threefold. First, we formally describe, model and solve the MMIRP exactly by branch-and-cut. We consider a large number of customers with several products, vehicles and planning horizon sizes so that the impact of each parameter can be properly measured. A secondary contribution is the introduction of consistency features in a multi-product environment, which is also assessed in our experiments. Third, we propose a large set of benchmark instances in order to evaluate our algorithm. These are made publicly available to the research community to allow the assessment of future algorithms. The proposed test bed is designed to cover a large set of combinations regarding the number of customers, the number of products, the number of vehicles and the length of the planning horizon, ranging from relatively easy instances to very challenging ones.

The remainder of the paper is organised as follows. In Section 2 we formally describe the MMIRP and the consistency features considered in this paper. In Section 3 we propose mixed-integer linear programming formulations for these problems. In Section 4 we present our branch-and-cut algorithm. We then provide the results of extensive computational experiments in Section 5. Conclusions follow in Section 6.

2. The multi-product multi-vehicle inventory-routing problem

We now formally introduce the MMIRP and the MMIRP with consistency features. These problems are defined on a graph $\mathcal{G} = (\mathcal{V}, \mathcal{A})$, where $\mathcal{V} = \{0, \dots, n\}$ is the vertex set and \mathcal{A} is the arc set. Vertex 0 represents the supplier and the vertices of $\mathcal{V}' = \mathcal{V} \setminus \{0\}$ represent customers. The supplier is responsible for distributing a set $\mathcal{M} = \{1, \dots, M\}$ of products to the customers. Both the supplier and customers incur a unit inventory holding cost h_i^m per period ($i \in \mathcal{V}, m \in \mathcal{M}$). The length of the planning horizon is p and, at each time period $t \in \mathcal{T} = \{1, \dots, p\}$, the quantity of product m made available at the supplier is r^{mt} . We assume the supplier has enough inventory to meet all the demand during the planning horizon and that inventories are not allowed to be negative. The variables I_0^{mt} and I_i^{mt} are defined as the inventory levels of product m at the end of period t , respectively at the supplier and at customer i . At the beginning of the planning horizon the decision maker knows the current inventory level of the supplier and of all customers (I_i^{m0} for $i \in \mathcal{V}, m \in \mathcal{M}$), and has full knowledge of the demand d_i^{mt} of product m of each customer i for each time period t . There is a set $\mathcal{K} = \{1, \dots, K\}$ of vehicles available. Each vehicle is able to perform one route per time period to deliver products from the supplier to a subset of customers. A routing cost c_{ij} is associated with arc $(i, j) \in \mathcal{A}$.

The objective of the MMIRP is to minimise the total inventory-distribution cost while meeting the demand for each customer. The replenishment plan is subject to the following constraints:

- the inventory level at each customer can never exceed its maximum capacity;
- inventory levels are not allowed to be negative;
- each of the supplier's vehicles can perform at most one route per time period; each route starts and ends at the supplier;
- vehicle capacities cannot be exceeded.

The solution to the problem should determine which customers to serve in each time period using which of the supplier's vehicles, how much to deliver of each product to each visited customer as well as which routes to use.

In order to increase quality of service, additional features may be added to reflect a number of concerns, e.g., workforce management (Francis, Smilowitz, and Tzur 2007; Zhong, Hall, and Dessouky 2007; Smilowitz, Nowak, and

Jiang 2012) and regularity of service (Coelho, Cordeau, and Laporte 2012b). To this end, Coelho, Cordeau, and Laporte (2012b) incorporated several consistency features in order to increase the quality of service while remaining cost-effective. Some of them are particularly meaningful when distributing several products. We incorporate the following two consistency features within the MMIRP.

- *Driver partial consistency*: Extending the work of Groër, Golden, and Wasil (2009) to the MMIRP, the standard driver consistency feature requires that each customer be assigned to only one driver. We implement a relaxation of this rule to allow some of the deliveries not to be subject to it, that is, most deliveries should be performed by the same driver to the same customer.
- *Visit spacing*: This feature imposes a temporal space between consecutive visits to the same customer.

3. Mathematical programming formulations

In this section we propose mixed-integer linear programming formulations for the MMIRP and for its counterpart with consistency features.

Assuming that the transportation cost matrix is symmetric, we work with an undirected formulation in order to reduce the number of variables. Thus, the model uses variables x_{ij}^{kt} equal to the number of times edge (i, j) with $i < j$ used on the route of vehicle k in period t . We also introduce binary variables y_i^{kt} equal to one if and only if node i (the supplier or a customer) is visited by vehicle k in period t . Let I_i^{mt} denote the inventory level of product m at vertex $i \in \mathcal{V}$ at the end of period $t \in \mathcal{T}$. Each customer $i \in \mathcal{V}'$ has a common inventory holding capacity C_i which is shared for all products $m \in \mathcal{M}$. The capacity of each vehicle is Q_k , which is also shared for all products being carried. We denote by q_i^{mkt} the quantity of product m delivered from the supplier using vehicle k to customer i in time period t . Note that with respect to the single product problem (Coelho and Laporte 2013), this model requires exactly the same number of binary variables. The only difference lies in the number of continuous variables I_i^{mt} and q_i^{mkt} . The problem can then be formulated as follows:

$$\text{minimise} \quad \sum_{i \in \mathcal{V}} \sum_{m \in \mathcal{M}} \sum_{t \in \mathcal{T}} h_i^m I_i^{mt} + \sum_{i \in \mathcal{V}} \sum_{j \in \mathcal{V}, i < j} \sum_{k \in \mathcal{K}} \sum_{t \in \mathcal{T}} c_{ij} x_{ij}^{kt}, \quad (1)$$

subject to

$$I_0^{mt} = I_0^{m,t-1} + r^{mt} - \sum_{k \in \mathcal{K}} \sum_{i \in \mathcal{V}'} q_i^{mkt}, \quad m \in \mathcal{M}, t \in \mathcal{T}, \quad (2)$$

$$I_i^{mt} = I_i^{m,t-1} + \sum_{k \in \mathcal{K}} q_i^{mkt} - d_i^{mt}, \quad m \in \mathcal{M}, i \in \mathcal{V}', t \in \mathcal{T}, \quad (3)$$

$$I_i^{mt} \geq 0, \quad i \in \mathcal{V}, m \in \mathcal{M}, t \in \mathcal{T}, \quad (4)$$

$$\sum_{m \in \mathcal{M}} I_i^{mt} \leq C_i, \quad i \in \mathcal{V}', t \in \mathcal{T}, \quad (5)$$

$$\sum_{m \in \mathcal{M}} \sum_{k \in \mathcal{K}} q_i^{mkt} \leq C_i - \sum_{m \in \mathcal{M}} I_i^{m,t-1}, \quad i \in \mathcal{V}', t \in \mathcal{T}, \quad (6)$$

$$q_i^{mkt} \leq C_i y_i^{kt}, \quad i \in \mathcal{V}', m \in \mathcal{M}, k \in \mathcal{K}, t \in \mathcal{T}, \quad (7)$$

$$\sum_{i \in \mathcal{V}'} \sum_{m \in \mathcal{M}} q_i^{mkt} \leq Q_k y_0^{kt}, \quad k \in \mathcal{K}, t \in \mathcal{T}, \quad (8)$$

$$\sum_{j \in \mathcal{V}, i < j} x_{ij}^{kt} + \sum_{j \in \mathcal{V}, j < i} x_{ji}^{kt} = 2y_i^{kt}, \quad i \in \mathcal{V}, k \in \mathcal{K}, t \in \mathcal{T}, \quad (9)$$

$$\sum_{i \in \mathcal{S}} \sum_{j \in \mathcal{S}, i < j} x_{ij}^{kt} \leq \sum_{i \in \mathcal{S}} y_i^{kt} - y_g^{kt}, \quad \mathcal{S} \subseteq \mathcal{V}', k \in \mathcal{K}, t \in \mathcal{T}, \quad (10)$$

for some $g \in \mathcal{S}$,

$$q_i^{mkt} \geq 0, \quad i \in \mathcal{V}', k \in \mathcal{K}, t \in \mathcal{T}, \quad (11)$$

$$x_{i0}^{kt} \in \{0, 1, 2\}, \quad i \in \mathcal{V}', k \in \mathcal{K}, t \in \mathcal{T}, \quad (12)$$

$$x_{ij}^{kt} \in \{0, 1\}, \quad i, j \in \mathcal{V}', k \in \mathcal{K}, t \in \mathcal{T}, \quad (13)$$

$$y_i^{kt} \in \{0, 1\}, \quad i \in \mathcal{V}, k \in \mathcal{K}, t \in \mathcal{T}. \quad (14)$$

Constraints (2) and (3) define the inventories at the supplier and at the customers, while constraints (4) prevent stockouts at the supplier. Constraints (5) impose maximal inventory level at the customers. Constraints (6) and (7) link the quantities delivered to the routing variables. In particular, they only allow a vehicle to deliver any products to a customer if the customer is visited by this vehicle. Constraints (8) ensure the vehicle capacities are respected while constraints (9) and (10) are degree constraints and subtour elimination constraints, respectively. Constraints (11)–(14) enforce integrality and non-negativity conditions on the variables.

3.1 Valid inequalities

The formulation just presented can be further strengthened by adding the following valid inequalities:

$$x_{i0}^{kt} \leq 2y_i^{kt}, \quad i \in \mathcal{V}, k \in \mathcal{K}, t \in \mathcal{T}, \quad (15)$$

$$x_{ij}^{kt} \leq y_i^{kt}, \quad i, j \in \mathcal{V}, k \in \mathcal{K}, t \in \mathcal{T}, \quad (16)$$

$$y_i^{kt} \leq y_0^{kt}, \quad i \in \mathcal{V}', k \in \mathcal{K}, t \in \mathcal{T}, \quad (17)$$

$$\sum_{k \in \mathcal{K}} \sum_{l=1}^t y_i^{kl} \geq \left\lceil \left(\sum_{k \in \mathcal{K}} \sum_{l=1}^{t-1} d_i^{mkl} - I_i^{m0} \right) / C_i \right\rceil, \quad i \in \mathcal{V}, m \in \mathcal{M}, t \in \mathcal{T}. \quad (18)$$

Constraints (15) and (16) are referred to as logical inequalities. They enforce the condition that if the supplier is the successor of a customer in the route of vehicle k in period t , i.e. $x_{i0}^{kt} = 1$ or 2 , then i must be visited by the same vehicle, i.e. $y_i^{kt} = 1$. A similar reasoning is applied to customer j in inequalities (16). Constraints (17) include the supplier in the route of vehicle k if any customer is visited by that vehicle in that period. Constraints (18) ensure that customer i is visited at least the number of times corresponding to the right-hand side of the inequality. This inequality is only valid if the fleet is homogeneous. It was originally developed for the single-vehicle case by Archetti et al. (2007) and was later extended to the multi-vehicle case by Coelho and Laporte (2013).

Finally, we also tighten this formulation by imposing the following symmetry-breaking constraints valid for the case where the vehicle fleet is homogeneous:

$$y_0^{kt} \leq y_0^{k-1,t}, \quad k \in \mathcal{K} \setminus \{1\}, t \in \mathcal{T}, \quad (19)$$

$$y_i^{kt} \leq \sum_{j < i} y_j^{k-1,t}, \quad i \in \mathcal{V}', k \in \mathcal{K} \setminus \{1\}, t \in \mathcal{T}. \quad (20)$$

Constraints (19) ensure that vehicle k cannot leave the depot if vehicle $k - 1$ is not used. This symmetry-breaking rule is then extended to the customer vertices by constraints (20), which state that if customer i is assigned to vehicle k in period t , then vehicle $k - 1$ must serve a customer with an index smaller than i in the same period. These constraints were also used by Coelho and Laporte (2013) and are inspired from those proposed by Fischetti, Salazar-González, and Toth (1995) for the capacitated vehicle routing problem and by Albareda-Sambola, Fernández, and Laporte (2011) for a plant location problem.

3.2 The MMIRP with consistency features

We now describe minor modifications to the model necessary to account for each of the consistency features described in Section 2.

3.2.1 Driver partial consistency

This feature can be handled in a number of ways. We have modelled it with an extra binary variable z_i^k equal to 1 if and only if vehicle k visits customer i . As in Coelho, Cordeau, and Laporte (2012b) we add to the objective function a penalty term proportional to the number of extra vehicles assigned to each customer, and we introduce a binary variable s_i^k indicating whether an extra vehicle k is assigned to customer i . We then impose the following sets of constraints to the model:

$$\sum_{k \in \mathcal{K}} z_i^k = 1, \quad i \in \mathcal{V}', \quad (21)$$

$$y_i^{kt} \leq z_i^k + s_i^k, \quad i \in \mathcal{V}', k \in \mathcal{K}, t \in \mathcal{T}, \quad (22)$$

$$s_i^k, z_i^k \in \{0, 1\}, \quad i \in \mathcal{V}', k \in \mathcal{K}. \quad (23)$$

Constraints (21) assign a first vehicle to each customer, while constraints (22) allow additional vehicles to be assigned to the same customer. We then add a penalty term,

$$\alpha \sum_{i \in \mathcal{V}'} \sum_{k \in \mathcal{K}} s_i^k, \quad (24)$$

to the objective function (1). By adjusting the parameter α , one can control how restrictive the driver partial consistency policy will be. If α is sufficiently large, the model considers only one driver per customer, or a strict driver consistency rule. If it is equal to zero, the model reduces to the standard MMIRP without consistency.

3.2.2 Visit spacing

Adding the following constraints to the basic model will ensure that at least one visit will take place every $(V_i + 1)$ periods, and no more than one visit will take place in any $(v_i + 1)$ successive periods:

$$\sum_{k \in \mathcal{K}} \sum_{l=t}^{t+v_i} y_i^{kl} \leq 1, \quad i \in \mathcal{V}', t \in \{1, \dots, p - v_i\}, \quad (25)$$

$$\sum_{k \in \mathcal{K}} \sum_{l=t}^{t+V_i} y_i^{kl} \geq 1, \quad i \in \mathcal{V}', \quad t \in \{1, \dots, p - V_i\}. \quad (26)$$

4. Branch-and-cut algorithm

The MMIRP is \mathcal{NP} -hard since it contains the Vehicle Routing Problem (VRP) as a special case. If the instance size is not excessive, the proposed undirected formulation can be solved exactly by branch-and-cut as follows. We provide a sketch of our branch-and-cut implementation in Algorithm 1.

4.1 Branch-and-cut scheme

At a generic node of the search tree, a linear program (LP) defined by (1)–(9) is solved, a search for violated subtour elimination constraints (10) is performed, and some of these constraints are added to the current program which is then re-optimised. This process is repeated until a feasible or dominated solution is reached, or until there are no more cuts to be added. At this point, branching on a fractional variable occurs. We accelerate the process of finding good feasible upper bounds by solving an exact mixed-integer program (MIP) as follows.

In order to solve the MMIRP with driver partial consistency, the objective function is the sum of (1) and (24), subject to (2)–(14). For the MMIRP with visit spacing consistency, the LP is defined by (1)–(9), (25) and (26). All valid inequalities remain the same.

4.2 Solution improvement algorithm

The purpose of the Solution Improvement (SI) algorithm is to approximate the cost of a new solution resulting from vertex removals and reinsertions. It is solved exactly whenever the branch-and-cut search identifies a new best solution. Using an idea proposed by Archetti et al. (2012), we simplify and approximate the routing costs resulting from vertex removals and reinsertions as follows. Let a_i^{kt} represent the routing cost reduction if customer i is removed from the route of vehicle k at period t ; let b_i^{kt} represent the routing cost if customer i is inserted in the route of vehicle k at period t ; finally, let r_i^{kt} be a binary parameter equal to 1 if and only if customer i is visited in the current route of vehicle k in period t . Also define the following binary variables: let u_i^{kt} be equal to 1 if and only if customer i is removed from the existing route of vehicle k in period t , and v_i^{kt} be equal to 1 if and only if customer i is inserted in the route of vehicle k in period t . This subproblem is then to

$$(SI) \quad \text{minimise} \quad \sum_{i \in \mathcal{V}'} \sum_{m \in \mathcal{M}} \sum_{t \in \mathcal{T}} h_i^m I_i^{mt} + \sum_{i \in \mathcal{V}'} \sum_{k \in \mathcal{K}} \sum_{t \in \mathcal{T}} (b_i^{kt} v_i^{kt} - a_i^{kt} u_i^{kt}), \quad (27)$$

subject to (2)–(6) and to

$$q_i^{mkt} \leq (r_i^{kt} - u_i^{kt} + v_i^{kt}) C_i, \quad i \in \mathcal{V}', \quad m \in \mathcal{M}, \quad k \in \mathcal{K}, \quad t \in \mathcal{T}, \quad (28)$$

$$v_i^{kt} \leq 1 - r_i^{kt}, \quad i \in \mathcal{V}', \quad k \in \mathcal{K}, \quad t \in \mathcal{T}, \quad (29)$$

$$u_i^{kt} \leq r_i^{kt}, \quad i \in \mathcal{V}', \quad k \in \mathcal{K}, \quad t \in \mathcal{T}, \quad (30)$$

$$\sum_{i \in \mathcal{V}'} u_i^{kt} + \sum_{i \in \mathcal{V}'} v_i^{kt} \leq \varepsilon, \quad k \in \mathcal{K}, \quad t \in \mathcal{T}, \quad (31)$$

$$\sum_{i \in \mathcal{V}'} \sum_{m \in \mathcal{M}} q_i^{mkt} \leq Q_k, \quad k \in \mathcal{K}, \quad t \in \mathcal{T}, \quad (32)$$

$$q_i^{mkt} \geq 0, \quad i \in \mathcal{V}', \quad m \in \mathcal{M}, \quad k \in \mathcal{K}, \quad t \in \mathcal{T}, \quad (33)$$

$$u_i^{kt}, v_i^{kt} \in \{0, 1\}, \quad i \in \mathcal{V}', k \in \mathcal{K}, t \in \mathcal{T}. \quad (34)$$

The objective function (27) minimises the total inventory, removal and insertion cost. Constraints (28) enforce the ML policy. Constraints (29) ensure that if a customer is already present in a route, it cannot be reinserted in the same route. Likewise, constraints (30) guarantee that only those customers belonging to a route can be removed from it. Constraints (32) ensure that the vehicle capacity is respected. If the incumbent solution is changed by more than one customer, then this model only provides an approximation of the actual routing costs. For this reason, we have decided to limit the number of insertions and removals that can take place in the solution of SI, and we have added constraints (31) to limit the number of insertions and removals per route to a small value ε .

4.2.1 SI for the driver partial consistency

The driver partial consistency is also modeled in SI with a binary variable s_i^k and a penalty in the objective function, as above. The variable s_i^k will be equal to one if and only if an extra vehicle k is assigned to customer i . The required constraints are

$$\sum_{k \in \mathcal{K}} z_i^k = 1, \quad i \in \mathcal{V}', k \in \mathcal{K}, t \in \mathcal{T}, \quad (35)$$

$$r_i^{kt} - u_i^{kt} + v_i^{kt} \leq z_i^k + s_i^k, \quad i \in \mathcal{V}', k \in \mathcal{K}, t \in \mathcal{T}, \quad (36)$$

$$s_i^k, z_i^k \in \{0, 1\}, \quad i \in \mathcal{V}', k \in \mathcal{K}. \quad (37)$$

The penalty to the objective function is added in the same fashion as in Section 3.2.1.

4.2.2 SI for the visit spacing consistency

The imposition of minimum and maximum intervals between visits is modelled by adding the following sets of constraints to the SI model:

$$\sum_{k \in \mathcal{K}} \sum_{l=t}^{t+m_i} (r_i^{kr} - u_i^{kl} + v_i^{kr}) \leq 1, \quad i \in \mathcal{V}', t \in \{1, \dots, p - v_i\}, \quad (38)$$

$$\sum_{k \in \mathcal{K}} \sum_{l=t}^{t+M_i} (r_i^{kr} - u_i^{kl} + v_i^{kr}) \geq 1, \quad i \in \mathcal{V}', t \in \{1, \dots, p - V_i\}. \quad (39)$$

Algorithm 1: Proposed branch-and-cut algorithm

- 1: Randomly generate a starting solution and apply SI to it.
- 2: At the root node of the search tree, generate and insert all valid inequalities (15)–(20) into the program.
- 3: Subproblem solution. Solve the LP relaxation of the node.
- 4: Termination check:
- 5: **if** there are no more nodes to evaluate **then**
- 6: Stop.
- 7: **else**
- 8: **if** The current solution is a new best solution **then**
- 9: Apply the SI algorithm to the incumbent solution.
- 10: **if** the SI algorithm yields an improved solution **then**
- 11: Update the solution vector at the branch-and-cut level
- 12: **end if**
- 13: **end if**
- 14: Select one node from the branch-and-bound tree.


```

15: end if
16: while the solution of the current LP relaxation contains subtours do
17:   Identify the connected components using the separation procedure of Padberg and Rinaldi (1991).
18:   Add all violated subtour elimination constraints (10).
19:   Subproblem solution. Solve the LP relaxation of the node.
20: end while
21: if the solution of the current LP relaxation is integer then
22:   Go to the termination check.
23: else
24:   Branching: branch on one of the fractional variables.
25:   Go to the termination check.
26: end if

```

5. Computational experiments

We now describe some details related to the computational experiments used to evaluate our algorithm. All computations were carried out on a grid of Intel XeonTM processors running at 2.66 GHz with up to 48 GB of RAM installed per node, with the Scientific Linux 6.1 operating system. A single thread was used. The algorithms just described were coded in C++ and we use IBM Concert Technology and CPLEX 12.4 as the MIP solver. The generation of the instances is described in Section 5.1 and detailed computational results are provided in Section 5.2.

5.1 Generation of the instances

We have generated a large data set varying the number of customers, products, vehicles and planning horizon. Our test bed is generated as follows:

- number of customers n : $10c$, where $c = 1, 2, 3, 4, 5$;
- number of products M : equal to 1, 3, or 5 products;
- number of vehicles K : equal to 1, 3, or 5 vehicles;
- horizon p : equal to 3, 5, or 7 periods;
- customer demand d_i^{mt} : drawn randomly from a discrete uniform distribution in the interval $[10, 100]$;
- product availability at the supplier r^{mt} : $f^{mt}n$, where f^{mt} is drawn randomly from a discrete uniform distribution in the interval $[50, 140]$;
- maximum inventory level C_i : $mg_i f_i$, where g_i is randomly selected from the set $\{2, 3, 4\}$ and f_i is drawn randomly from a discrete uniform distribution in the interval $[150, 200]$;
- starting inventory level I_i^{m0} : drawn randomly from a discrete uniform distribution in the interval $[100, 150]$;
- starting inventory level I_0^{m0} : $\sum_i I_i^{m0}$;
- inventory holding cost h_0^m : 0.01;
- inventory holding cost h_i^m ($i > 0$): drawn randomly from a continuous uniform distribution in the interval $[0.02, 0.20]$;
- vehicle capacity Q : $2 \lfloor \sum_i \sum_m \sum_t d_i^{mt} / pK \rfloor$;
- distance/cost c_{ij} : $\lfloor \sqrt{(X_i - X_j)^2 + (Y_i - Y_j)^2} + 0.5 \rfloor$, where the points (X_i, Y_i) are the coordinates of vertex i and are obtained randomly from a discrete uniform distribution in the interval $[0, 1000]$.

For each combination of n , M , K , and p we generated five instances, yielding a total of 675 instances. This set of instances as well as the solutions presented below are available at <http://www.leandro-coelho.com/instances/>.

5.2 Computational experiments

We now report solutions for the MMIRP without consistency features in Section 5.2.1 and for the MMIRP with consistency features in Section 5.2.2.

Table 1. Summary of computational results for the MMIRP with one product ($M = 1$).

Horizon	# customers	$K = 1$			$K = 3$			$K = 5$		
		Gap (%)	# solved	Time (s)	Gap (%)	# solved	Time (s)	Gap (%)	# solved	Time (s)
$p = 3$	10	0.00	5	0.00	0.00	5	6.60	0.00	5	120.20
	20	0.00	5	1.60	0.00	5	184.00	4.09	2	29,770.40
	30	0.00	5	13.20	0.00	5	5113.40	15.53	0	43,200.00
	40	0.00	5	101.80	2.92	3	28,221.60	16.20	0	43,189.80
	50	0.00	5	299.60	19.44	0	43,200.00	32.22	0	43,189.00
$p = 5$	10	0.00	5	1.00	0.22	4	12,555.40	12.82	1	35,353.80
	20	0.00	5	20.40	17.04	0	41,683.20	43.33	0	43,200.00
	30	0.00	5	182.20	31.16	0	43,181.00	43.84	0	43,197.40
	40	0.00	4	9464.60	42.11	0	43,188.80	51.82	0	43,197.60
	50	0.56	4	20,712.60	48.12	0	43,197.20	65.98	0	43,200.00
$p = 7$	10	0.00	5	27.80	14.40	0	28,173.20	27.67	0	40,106.40
	20	0.00	5	3335.60	27.67	0	41,297.40	46.27	0	43,175.40
	30	3.06	2	28,529.00	33.60	0	43,171.40	61.35	0	43,191.20
	40	6.03	0	43,200.00	61.12	0	43,200.00	71.09	0	43,200.00
	50	9.97	0	43,187.80	62.17	0	43,200.00	71.61	0	40,688.40

Table 2. Summary of computational results for the MMIRP with three products ($M = 3$).

Horizon	# customers	$K = 1$			$K = 3$			$K = 5$		
		Gap (%)	# solved	Time (s)	Gap (%)	# solved	Time (s)	Gap (%)	# solved	Time (s)
$p = 3$	10	0.00	5	1.60	0.00	5	5.80	0.00	5	102.40
	20	0.00	5	0.80	0.00	5	322.60	2.46	3	21,424.00
	30	0.00	5	1.00	0.00	5	3574.60	3.81	1	36,036.80
	40	0.00	5	5.20	0.26	4	12,952.60	2.36	1	42,241.80
	50	0.00	5	11.40	2.27	2	35,501.60	9.46	0	43,200.00
$p = 5$	10	0.00	5	3.40	0.00	5	8008.20	19.39	0	41,231.80
	20	0.00	5	45.80	17.05	0	42,202.40	26.03	0	43,200.00
	30	0.00	5	506.20	20.67	0	43,196.00	37.85	0	43,181.00
	40	0.00	5	6967.80	23.57	0	43,188.60	51.09	0	37,073.80
	50	2.72	1	40,242.60	45.54	0	18,555.20	47.08	0	21,927.60
$p = 7$	10	0.00	5	23.80	21.94	0	27,184.00	30.18	0	42,541.20
	20	0.00	5	6173.40	29.36	0	43,197.60	41.00	0	43,195.20
	30	5.41	0	43,200.00	30.89	0	43,197.80	60.65	0	43,190.00
	40	11.04	0	43,200.00	38.35	0	43,200.00	61.09	0	34,766.60
	50	15.15	0	43,199.80	54.69	0	43,200.00	56.43	0	43,200.00

Table 3. Summary of computational results for the MMIRP with five products ($M = 5$).

Horizon	# customers	$K = 1$			$K = 3$			$K = 5$		
		Gap (%)	# solved	Time (s)	Gap (%)	# solved	Time (s)	Gap (%)	# solved	Time (s)
$p = 3$	10	0.00	5	4.60	0.00	5	5.00	0.00	5	64.00
	20	0.00	5	1.00	0.00	5	520.00	0.60	4	9813.20
	30	0.00	5	2.00	0.81	3	17,738.80	3.13	2	37,412.20
	40	0.00	5	6.40	0.71	4	14,205.40	5.55	0	43,199.40
	50	0.00	5	19.00	1.14	2	33,966.20	12.14	0	41,009.60
$p = 5$	10	0.00	5	7.40	5.10	3	16,428.00	17.73	0	43,122.00
	20	0.00	5	95.20	15.28	0	43,200.00	30.16	0	43,200.00
	30	0.00	5	5670.40	22.49	0	43,188.60	39.07	0	43,200.00
	40	2.11	2	32,088.60	30.40	0	43,194.60	54.65	0	43,200.00
	50	7.49	0	43,194.00	45.56	0	42,173.00	54.34	0	41,344.60
$p = 7$	10	0.00	5	53.60	22.77	0	26,783.20	29.57	0	42,114.00
	20	0.33	4	18,773.60	30.34	0	43,200.00	40.03	0	43,200.00
	30	9.39	0	43,195.00	30.84	0	43,200.00	44.05	0	43,185.40
	40	12.30	0	43,200.00	62.96	0	43,192.00	54.70	0	43,200.00
	50	17.64	0	43,194.80	55.65	0	43,200.00	55.06	0	43,200.00

5.2.1 Computational results for the MMIRP

We organise the presentation of the computational results based on the number of products, which is the main feature of this paper. We compare the solutions obtained for different combinations of the parameters. Specifically, we present in Tables 1–3 average solution values for all combinations of the remaining parameters, namely the time horizon $p = 1, 3$, and 5, and the number of vehicle $K = 1, 3$, and 5. In each table we present averages comprising the five instances for each number of customers. We show the average percentage gap, the number of instances solved to optimality and the average running time.

Table 1 shows that the average size of instances solved to optimality are similar to those of previous studies on the single product IRP (Adulyasak, Cordeau, and Jans 2012; Coelho and Laporte 2013). Comparing these figures with those of Tables 2 and 3 indicates that the addition of more products to the problem does not increase its difficulty. This is because the number of binary variables remains unchanged if more products are added to the problem. For instance, for a problem with 50 customers, one product, five vehicles and seven time periods, the number of binary variables is 41,740 and the number of general variables is slightly over 15,000. For the same instance with five products, approximately 10,000 continuous variables are added to the problem, but no extra binary variable are needed. This explains the relative ease of solving problems with multiple products. For the sake of comparison, for this instance size, each time period requires approximately 8000 binary variables and each vehicle requires approximately 10,000 binary variables.

5.2.2 Computational results for the MMIRP with consistency

We also solved a subset of instances with the two consistency features presented in Sections 3.2.1 and 3.2.2. We applied these consistency features to the instances with 30 customers, and all combinations of three and five periods, vehicles, and products. These instances are significantly difficult since many of them could not be solved to optimality in the case without consistency. We compare the solution cost of each set of instances with respect to the base case without consistency requirements. Specifically, in Tables 4 and 5 we provide the average gap after solving these instances with a 12-hour time limit, the average running time and the average cost increase of each set of instances with respect to the solutions obtained without consistency features presented in the previous section.

In line with the observations of Coelho and Laporte (2013), ensuring the driver partial consistency feature does not increase the solution cost by much. It remains a meaningful way of increasing the quality of the solutions provided by the IRP both to customers and suppliers in a multi-product environment. The visit spacing consistency feature has shown, once again and for the first time in a multi-product framework, that it helps reduce the search space while

Table 4. Summary of computational results for the MMIRP with driver partial consistency ($n = 30$).

Products	Horizon	$K = 3$		
		Gap (%)	Time (s)	% increase
$M = 3$	$p = 3$	3.91	43,200.0	0.07
	$p = 5$	28.12	43,185.2	6.17
$M = 5$	$p = 3$	2.63	30,200.2	6.71
	$p = 5$	25.24	43,181.0	1.75

Table 5. Summary of computational results for the MMIRP with visit spacing consistency ($n = 30$).

Products	Horizon	$K = 3$		
		Gap (%)	Time (s)	% increase
$M = 3$	$p = 3$	0.00	662.8	0.00
	$p = 5$	21.33	43,200.0	3.85
$M = 5$	$p = 3$	0.00	891.6	−0.10
	$p = 5$	17.67	43,200.0	−2.84

providing meaningful solutions. Indeed, depending on the values chosen for its intervals the solutions it provides are valid for the most general case without consistency, but the introduction of this feature significantly reduces the time needed to obtain such good solutions. This is the case, for example, when solving instances with five products and five periods, which are very difficult for the general case, but for which the MMIRP with visit spacing consistency was able to find better solutions. This shows that some of these consistency features are beneficial not only from a business perspective but also from a computational point of view.

6. Conclusions

We have developed for the first time a simple yet powerful branch-and-cut algorithm to solve the IRP with multi-products and multi-vehicles. Our formulation is flexible and our algorithm is able to solve the IRP with one or several vehicles, and with many products, each with a specific demand, but sharing inventory and vehicle capacities. We have also solved instances' consistency features by imposing some regularity in the distribution process. We have proposed a large set of benchmark instances ranging from 10 to 50 customers, with up to seven time periods, five vehicles and five products for which we confirm the success of our exact algorithm.

Acknowledgements

This work was partly supported by the Canadian Natural Sciences and Engineering Research Council under grant 39682-10. This support is gratefully acknowledged. We also thank the RQCHP (Réseau québécois de calcul de haute performance) for providing computing facilities.

References

- Adulyasak, Y., J.-F. Cordeau, and R. Jans. 2012. *Formulations and branch-and-cut algorithms for multi-vehicle production and inventory routing problems*. Technical Report G-2012-14, GERAD, Montreal, Canada.
- Albareda-Sambola, M., E. Fernández, and G. Laporte. 2011. "A computational comparison of several models for the exact solution of the capacity and distance constrained plant location problem." *Computers & Operations Research* 38 (8): 1109–1116.
- Alegre, J., M. Laguna, and J. Pacheco. 2007. "Optimizing the periodic pick-up of raw materials for a manufacturer of auto parts." *European Journal of Operational Research* 179 (3): 736–746.
- Archetti, C., et al. 2007. "A branch-and-cut algorithm for a vendor-managed inventory-routing problem." *Transportation Science* 41 (3): 382–391.
- Archetti, C., et al. 2012. "A hybrid heuristic for an inventory routing problem." *INFORMS Journal on Computing* 24 (1): 101–116.
- Bausch, D.O., G.G. Brown, and D. Ronen. 1998. "Scheduling short-term marine transport of bulk products." *Maritime Policy & Management* 25 (4): 335–348.
- Bell, W.J., et al. 1983. "Improving the distribution of industrial gases with an on-line computerized routing and scheduling optimizer." *Interfaces* 13 (6): 4–23.
- Bertazzi, L., and M.G. Speranza. 2002. "Continuous and discrete shipping strategies for the single link problem." *Transportation Science* 36 (3): 314–325.
- Bertazzi, L., M.G. Speranza, and W. Ukovich. 1997. "Minimization of logistic costs with given frequencies." *Transportation Research Part B: Methodological* 31 (4): 327–340.
- Bertazzi, L., G. Paletta, and M.G. Speranza. 2002. "Deterministic order-up-to level policies in an inventory routing problem." *Transportation Science* 36 (1): 119–132.
- Carter, M.W., et al. 1996. "Solving an integrated logistics problem arising in grocery distribution." *INFOR* 34 (4): 290–306.
- Chien, T.W., A. Balakrishnan, and R.T. Wong. 1989. "An integrated inventory allocation and vehicle routing problem." *Transportation Science* 23 (2): 67–76.
- Christiansen, M., et al. 2011. "Maritime inventory routing with multiple products: A case study from the cement industry." *European Journal of Operational Research* 208 (1): 86–94.
- Coelho, L.C., and G. Laporte. 2013. "The exact solution of several classes of inventory-routing problems." *Computers & Operations Research* 40 (2): 558–565.
- Coelho, L.C., J.-F. Cordeau, and G. Laporte. 2012a. "The inventory-routing problem with transshipment." *Computers & Operations Research* 39 (11): 2537–2548.
- Coelho, L.C., J.-F. Cordeau, and G. Laporte. 2012b. "Consistency in multi-vehicle inventory-routing." *Transportation Research Part C: Emerging Technologies* 24 (1): 270–287.
- Engineer, F.G., et al. 2012. "A branch-and-price-and-cut algorithm for single-product maritime inventory routing." *Operations Research* 60 (1): 106–122.
- Federgruen, A., and P.H. Zipkin. 1984. "A combined vehicle-routing and inventory allocation problem." *Operations Research* 32 (5): 1019–1037.

- Federgruen, A., G. Prastacos, and P.H. Zipkin. 1986. "An allocation and distribution model for perishable products." *Operations Research* 34 (1): 75–82.
- Fischetti, M., Salazar-González, J.J., and Toth, P., 1995. Experiments with a multi-commodity formulation for the symmetric capacitated vehicle routing problem. In: *Proceedings of the 3rd meeting of the EURO working group on transportation*, Barcelona, Spain, 169–173.
- Francis, P., K. Smilowitz, and M. Tzur. 2007. "Flexibility and complexity in periodic distribution problems." *Naval Research Logistics* 54 (2): 136–150.
- Groër, C., B.L. Golden, and E.A. Wasil. 2009. "The consistent vehicle routing problem." *Manufacturing & Service Operations Management* 11 (4): 630–643.
- Grønhaug, R., et al. 2010. "A branch-and-price method for a liquefied natural gas inventory routing problem." *Transportation Science* 44 (3): 400–415.
- Huang, S.-H., and P.-C. Lin. 2010. "A modified ant colony optimization algorithm for multi-item inventory routing problems with demand uncertainty." *Transportation Research Part E: Logistics and Transportation Review* 46 (5): 598–611.
- Moin, N.H., S. Salhi, and N.A.B. Aziz. 2011. "An efficient hybrid genetic algorithm for the multi-product multi-period inventory routing problem." *International Journal of Production Economics* 133 (1): 334–343.
- Padberg, M.W., and G. Rinaldi. 1991. "A branch-and-cut algorithm for the resolution of large-scale symmetric traveling salesman problems." *SIAM Review* 33 (1): 60–100.
- Persson, J.A., and M. Göthe-Lundgren. 2005. "Shipment planning at oil refineries using column generation and valid inequalities." *European Journal of Operational Research* 163 (3): 631–652.
- Popović, D., M. Vidović, and G. Radivojević. 2012. "Variable neighborhood search heuristic for the inventory routing problem in fuel delivery." *Expert Systems with Applications* 39 (18): 13390–13398.
- Qu, W.W., J.H. Bookbinder, and P. Iyogun. 1999. "An integrated inventory-transportation system with modified periodic policy for multiple products." *European Journal of Operational Research* 115 (2): 254–269.
- Ramkumar, N. et al., 2012. "Mixed integer linear programming model for multi-commodity multi-depot inventory routing problem." *OPSEARCH* 49 (4): 413–429.
- Ronen, D. 2002. "Marine inventory routing: Shipments planning." *Journal of the Operational Research Society* 53 (1): 108–114.
- Sindhuchao, S., et al. 2005. "An integrated inventory-routing system for multi-item joint replenishment with limited vehicle capacity." *Journal of Global Optimization* 32 (1): 93–118.
- Smilowitz, K., M. Nowak, and T. Jiang. 2012. "Workforce management in periodic delivery operations." *Transportation Science*, forthcoming (doi: 10.1287/trsc.1120.0407).
- Solyal, O., and 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.
- Speranza, M.G., and W. Ukovich. 1994. "Minimizing transportation and inventory costs for several products on a single link." *Operations Research* 42 (5): 879–894.
- Speranza, M.G., and W. Ukovich. 1996. "An algorithm for optimal shipments with given frequencies." *Naval Research Logistics* 43 (5): 655–671.
- Stålhane, M., et al. 2012. "A construction and improvement heuristic for a liquefied natural gas inventory routing problem." *Computers & Industrial Engineering* 62 (1): 245–255.
- Uggen, K.T., M. Fodstad, and V.S. Nørstebø. 2011. "Using and extending fix-and-relax to solve maritime inventory routing problems". *TOP*, forthcoming (doi: 10.1007/s11750-011-0174-z).
- Zhong, H., R.W. Hall, and M.M. Dessouky. 2007. "Territory planning and vehicle dispatching with driver learning." *Transportation Science* 41 (1): 74–89.