Available online at www.sciencedirect.com

ScienceDirect

Electronic Notes in DISCRETE MATHEMATICS

Electronic Notes in Discrete Mathematics 52 (2016) 197–204 www.elsevier.com/locate/endm

A Hybrid Heuristic based on Iterated Local Search for Multivehicle Inventory Routing Problem

Edcarllos Santos [a], Luiz Satoru Ochi [a], Luidi Simonetti [a] and Pedro Henrique González [a,b]

[a] *Instituto de Computação, Universidade Federal Fluminense, Niterói, Brasil*

[b] *Laboratoire d'Informatique d'Avignon, Université d'Avignon et des Pays de Avignon, France*

Abstract

We study a multivehicle inventory routing problem (MIRP) in which supplier delivers one type of product along a finite planning horizon, using a homogeneous fleet of vehicles. The main objective is to minimize the total cost of storage and transportation. In order to solve MIRP, we propose an algorithm based on iterated local search (ILS) metaheuristic, using a variable neighborhood descent with random neighborhood ordering (RVND) in the local search phase. Moreover, we combined this algorithm with an exact procedure based on mathematical programming to solve specifically the inventory management as a subproblem. To validate our approach, computational tests were performed on 560 benchmark instances, achieving very competitive results in comparison to the best known algorithms.

Keywords: Multivehicle Inventory Routing Problem, Hybrid Metaheuristics, ILS.

1 Introduction

Over the last years, logistics became an important market trend providing efficiency and control over all stages of the supply chain. In this sense arises the Inventory Routing Problem (IRP) in which decisions acts in minimization of operational costs over distribution and storage of products. In this paper, we study an extension of IRP called Multivehicle Inventory Routing Problem (MIRP), which considers a homogeneous fleet of vehicles to distribute a single product in a finite horizon from a supplier to several customers.

http://dx.doi.org/10.1016/j.endm.2016.03.026

To solve this problem, different approaches have been proposed. In [2] the problem was modeled and a hybrid heuristic based in Adaptive Large Neighborhood Search (ALNS) which combines the exact solution from two MILP's was developed. To improve the quality of service, authors adopted structural policies such as maximum level and order-up-to. [1] proposed new formulations and branch-and-cut algorithms for two versions of IRP, using symmetry breaking constraints to improve the formulations and an ALNS heuristic to determine upper bounds. In [3] was presented a branch-and-cut algorithm to deal with different classes of MIRP's. In addition, some inequalities were adapted to support multivehicle and a heuristic was developed to improve the quality of the solutions found in search tree. Finally, in [4] was presented a new formulation and a branch-and-price-and-cut algorithm for MIRP, applying an ad-hoc labeling heuristic to solve column generation subproblems.

This paper proposes a hybrid multi-start algorithm based on ILS, using a RVND procedure in local search phase. Furthermore, an exact procedure is used to solve the inventory management subproblem and a new constructive heuristic is presented. This paper is organized as follow. In Section 2, we give a formal definition for MIRP. In Section 3, we describe the solution approaches to solve the problem. Computational experiments are reported in Section 4. Finally, Section 5 presents the concluding remarks of this work.

2 Problem definition

Let $G = (V, E)$ denote a graph where V is the set of vertices and E is the set of edges. Vertex 0 represents the supplier and $V' = V \backslash \{0\}$ the customers. Demand of customer $v \in V'$ in period $t \in T$ is represented by d_v^t. Supplier produces p_t units of product in period t, in such way its inventory is able to supply all demands in planning horizon. Each customer $v \in V'$ keeps an inventory with maximum capacity C_v. To meet all demands, supplier has a homogeneous fleet composed by K vehicles of capacity Q, which perform at most one route per period.

Quantity of product delivered from supplier to customer v in period t is denoted by variable q_v^t. The inventory level at end of period t for a supplier or customer v is represented by variable I_v^t, where for each unit of product stored in inventory, there is an inventory cost h_v associated ($v \in V, t \in T$). A routing cost associated to each edge $(i, j) \in E$ is defined by c_{ij}. Thus, the main objective is to minimize operational costs regarding to transportation and storage. In MIRP the following decisions must be noticed: when a customer must be served; how many units of products should be delivered in each replenishment service; what are the best routes for distribution. These decisions are subject to the following constraints: each route starts and ends at the supplier; no route exceeds the vehicle capacity; no stockout is allowed; storage maximum capacity of each customer can never be exceed. More details about mathematical formulation for MIRP can be found in [2].

3 Solution approach

Iterated local search (ILS) is a global optimization method that explores the solution space by successive perturbations from local optima solutions. These solutions are obtained in local search phase, which in turn uses as initial point a solution provided from a constructive procedure or perturbation mechanism. Thus, instead of considering the entire solution space, ILS focuses its search just on a reduced set composed by local optima. All components required to develop an ILS algorithm are described in next section.

3.1 Constructing the initial solution

Constructive heuristic, called *Forward Delivery*, is able to generate good quality solutions through decoupling of routing and inventory decisions. Sequentially, an analysis in two stages is performed for each period of the problem.

First stage is described by Procedure 1 and consists of supply, if necessary, the demand for a specific period through a new service. Given a period t, the feasibility of inventory level is checked for each customer $v \in V$. Case a stockout be identified (line 2), a new replenishment service is created to supply the minimum demand required to avoid the infeasibility on period t (line 3) and customer v is added to a list LC (line 4). Finally, all customers in list LC are routed on period t (line 6). Two insertion criteria are used to determine the cost of inserting an unrouted customer $k \in LC$ in a given route r on period t. The first one, denoted by $g(k,t) = c^r_{ik}$, computes the distance between a customer k and every customer i already included in partial solution. The second criteria, denoted by $g(k,t) = (c^r_{ik} + c^r_{kj} - c^r_{ij}) - \gamma(c^r_{0k} + c^r_{k0})$, determines the insertion cost between the client k and each pair of adjacent customers i and j already included in partial solution. In this case, the second term corresponds to a penalty to avoid late insertions of customers located far away from supplier. In both criteria, the insertion associated with the least-cost is done and then the customer routed is removed from list LC.

Second stage is presented in Procedure 2 and refers to expand quantity of periods served by a replenishment service already created in first stage, through increasing amount of product delivered to customer at period t. A period t_f greater than or equal t is randomly chosen in order to diversify the initial solution (line 3). If inventory is capable of storing minimum amount necessary to meet the demand of interval (line 13), then inventory is updated adding this new amount to the quantity of delivery on period t for customer v (line 16). Otherwise, interval is reduced (line 11) and the analysis is restarted. Finally, the residual capacity of associated route for customer v is updated (line 17). Both stages run sequentially for each period of the planning horizon.

Procedure 1: Phase 1

```
Input: Solution s, Period t, LC ← ∅
foreach Customer v ∈ V do
    if I_v^t < 0 then
        UpdateInventory(v, t, −I_v^t)
        LC ← LC + {v}
LC ←Shuffle(LC)
s ← BuildRoutes(LC, t)
return s, t, LC
```

Algorithm 1: HILS-RVND

```
f(s*) ← ∞
for iterMS ← 0 until MaxIterMS do
    iterILS ← 0
    s ← ConstructInitialSolution()
    for iterILS ← 0 until MaxIterILS do
        s ← localSearch(s)
        if f(s) < f(s*) then
            s* ← s
            iterILS ← −1
        s ← Perturbation(s*)
        iterILS ← iterILS + 1
    iterMS ← iterMS + 1
s ← InventoryManagement(s*)
```

Procedure 2: Phase 2

```
   Input: Solution s, Period t, List LC
1  LC ←Shuffle(LC)
2  foreach Customer v ∈ LC do
3      t_f ← Random(t, T)
4      if t_f = t then
5          v ← v + 1
6      while t_f > t do
7          del ← 0
8          if I_v^{t_f} < 0 then
9              del ← del − I_v^{t_f}
10         if operation is not feasible then
11             t_f ← t_f − 1
12         else
13             end while
14     if operation is feasible then
15         if I_v^{t_f} < 0 then
16             UpdateInventory(v, t, −I_v^{t_f})
17         s ←UpdateRoute(del)
18 return s
```

3.2 Local Search

Four inter-period neighborhood structures were developed to perform an integrated analysis of routing and inventory decisions simultaneously.

Service Insertion: A new replenishment service is created in period t_i for customer v, through relocation of products related to a service in period t_r. Thus, routing cost is increased and, in return, inventory cost is decreased. This movement is subject to the following preconditions: $t_r < t_i$; $q_v^{t_r} > 0$; $q_v^{t_i} = 0$; some route with positive residual capacity must exist in period t_i.

Service Removal: A replenishment service is removed in period t_r for customer v and its delivery is shifted to an existing service in period t_i. Exchange performed reduces routing cost and, in return, increases inventory cost. This operation is subject to the following requirements: $q_v^{t_r} > 0$; $q_v^{t_i} > 0$; route associated to customer in period t_i, must have residual capacity enough to store all products from removed replenishment service in period t_r.

Shift Delivery: Part of a delivery for customer v from a service in period t_r is shifted to an existing service in period t_i. In this way, just inventory cost is decreased. This movement is subject to the following preconditions: $t_r < t_i$; $q_v^{t_r} > 0$; some route with positive residual capacity must exist in period t_i.

Swap Route: Transfer a complete route rt from period t_r to period t_i, decreasing inventory cost and keeping unchanged the routing cost. This operation is subject to the following requirements: $t_r < t_i$; $\sum_{v \in rt} q_v^{t_i} = 0$; some unused vehicle must exist in period t_i.

Aiming to reduce transportation costs between routes of same period, five inter-route neighborhood structures were implemented: *Swap(1,1)*, *Swap(2,1)*, *Swap(2,2)*, *Shift(1,0)* and *Shift(2,0)*. Furthermore, five intra-route neighborhood structures were implemented in order to reduce transportation cost for each route: *Exchange*, *2-opt*, *Reinsertion*, *Or-opt2* and *Or-opt3*. More details about these classical neighborhood structures can be found in [5].

3.3 Inventory management

In order to decrease inventory costs, we propose a linear programming model able to determine the best solution for inventory management subproblem, given routes as an input solution. Let binary constant z_v^t be equal to 1, if and only if, there a replenishment service for customer v and period t in current solution used as input. This problem can be formulated as follows:

$$\text{Minimize} \qquad \sum_{v \in V} \sum_{t \in T} h_v I_v^t \tag{1}$$

Subject to:

$$I_0^t = I_0^{t-1} + p^t - \sum_{v \in V'} q_v^t \qquad t \in T \tag{2}$$

$$I_v^t = I_v^{t-1} + q_v^t - d_v^t \qquad v \in V' \quad t \in T \tag{3}$$

$$I_v^t \geq 0 \qquad v \in V \quad t \in T \tag{4}$$

$$I_v^t \leq C_v \qquad v \in V' \quad t \in T \tag{5}$$

$$q_v^t \leq C_v - I_v^{t-1} \qquad v \in V' \quad t \in T \tag{6}$$

$$\sum_{v \in rt} q_v^t \leq Q \qquad t \in T \tag{7}$$

$$q_v^t \geq z_v^t \qquad v \in V' \quad t \in T \tag{8}$$

$$q_v^t = 0 \qquad v \in V' \quad t \in T, \quad \forall\, z_v^t = 0 \tag{9}$$

Objective function (1) is the sum of inventory costs for supplier and customers. Constraints (2) and (3) determine, respectively, inventory level for supplier and customers. Constraints (4) and (5) allow a minimum and maximum inventory level. Constraints (6) ensure inventory level for a customer after a replenishment service cannot exceed maximum capacity. Constraints

(7) guarantee sum of products delivered for all customers served by route rt don't exceed the vehicle capacity. Finally, constraints (8) ensure, for each period, a minimum delivery quantity for customers served by a route.

3.4 HILS-RVND

All ILS procedures previously presented are incorporated into HILS-RVND heuristic, shown in Algorithm 1. At first, for each multi-start iteration a solution is built by forward delivery constructive heuristic (line 4). After, local search is performed through successive applications of RVND procedures composed exclusively by each type of neighborhood structures (line 6). Movements are applied exhaustively using best improvement as acceptance criteria and just feasible solutions are taken into account. Further, an inter-period structure is randomly selected and then multiple movements are arbitrarily performed (line 10). The number of movements applied in perturbation phase is chosen at random within the discrete interval $\{1, 2, 3\}$. Finally, exact solution for inventory management subproblem is embed to current solution through solving of model proposed in this paper (line 13).

4 Computational results

Algorithms in this paper were implemented in C++ and IBM Cplex Concert Technology 12.2 was utilized as solver for inventory management subproblem. Computational tests were performed on a Intel®Core™2 Quad 2.83 GHz PC with 8 GB of RAM running Ubuntu 12.04 OS and just one thread was used.

To evaluate the performance, proposed algorithm was tested in a data set introduced in [2], composed by 560 instances. Data set is split in two subgroups. First one presents a planning horizon $|T| \in \{3\}$ and $|V| \in \{10, 15, 20, 25, 30, 35, 40, 45, 50\}$ customers. Second subgroup consists of a planning horizon $|T| \in \{6\}$ and $|V| \in \{10, 15, 20, 25, 30\}$ customers. Both subgroups have a homogeneous fleet composed by $|K| \in \{2, 3, 4, 5\}$ vehicles and each combination has a low and high inventory holding instance and five different configurations. By calibration tests, parameters $MaxIterMS$ and $MaxIterILS$ were respectively fixed at $(10000 * T * K)$ and 10. For each instance, HILS-RVND was performed 10 times with a time limit of 300 seconds.

In following tables, quality indicators were used to classify our solutions over other approaches. Columns *Ind*, *Quantity* and *Optimal* are, respectively, the quality indicators, the quantity of average solutions and the amount of optimal solutions achieved. Column *Avg Gap* is the gap between our average

		HILS-RVND					Exacts[a]
|K|	Ind	Quantity	Quantity (%)	Optimal	Avg Gap (%)	Avg Time	Time
2	Better	6	4,29	-	-9,53	-	-
	Draw	40	28,57	40	0,00	-	-
	Worse	94	67,14	-	1,72	-	-
	Subtotal	140	100,00	40	0,78	50,49	1.641,89
3	Better	28	20,00	-	-1,95	-	-
	Draw	51	36,43	48	0,00	-	-
	Worse	61	43,57	-	1,77	-	-
	Subtotal	140	100,00	48	0,42	86,32	4.094,01
4	Better	42	30,00	-	-7,75	-	-
	Draw	19	13,57	19	0,00	-	-
	Worse	79	56,43	-	2,14	-	-
	Subtotal	140	100,00	19	-1,10	124,43	4.289,72
5	Better	49	35,00	-	-14,02	-	-
	Draw	17	12,14	17	0,00	-	-
	Worse	74	52,86	-	3,32	-	-
	Subtotal	140	100,00	17	-3,13	155,51	4.366,78
All	Better	125	22,32	-	-8,99	-	-
	Draw	127	22,68	124	0,00	-	-
	Worse	308	55,00	-	2,22	-	-
	Total	560	100,00	124	-0,76	104,19	3.598,10

a - Desaulniers et al [4], Intel® Core™ i7 3.40 GHz with 16 GB of RAM with a time limit of 7200 seconds

Table 1: Summary of results compared to exact approaches

solutions and best known solutions for another approach. Finally, column *Avg Time* is the average running time in seconds.

Table 1 presents results of HILS-RVND compared to best solutions of branch-and-cut [3] and branch-price-and-cut [4] algorithms. Results show HILS-RVND was able to match in 127 instances the best known solutions, within these, 124 with proven optimality. In addition, 125 another solutions were improved. Average gap was -0.76% and with respect to computational effort, average time was about 34 times smaller than exact approaches. Table 2 shows our results compared to a hybrid ALNS presented in [2]. Values shown in [2] consist of average values of solutions obtained for five instances, so comparison had to be performed separately. In this case, HILS-RVND improved 32 solutions, which corresponds to 66% of instances. Further, average gap was -0.33% and average running time was about 64 times lower than ALNS.

5 Final remarks

A new method to solve multivehicle inventory routing problem was proposed. Method consists of a hybrid heuristic based on iterated local search metaheuristic, using a variable neighborhood descent with random neighborhood

	HILS-RVND				ALNS[a]
Ind	**Quantity**	**Quantity (%)**	**Avg Gap (%)**	**Avg Time (s)**	**Avg Time (s)**
Better	32	66,66	-0,76	-	-
Draw	0	0,00	0,00	-	-
Worse	16	33,33	0,51	-	-
Total	48	100,00	-0,33	77,40	4.934,69

a - Coelho et al. [2], AMD Dual Core Opteron 2.20 GHz

Table 2: Summary of results compared to hybrid ALNS

ordering in local search phase. To generate initial solutions, a new constructive heuristic was developed in which decomposes decisions regarding to distribution and storage of products. Further, an exact procedure was used in order to deal with inventory management subproblem. Our experiments show that the proposed method was capable to find good quality solutions when compared to other approaches in literature. Another point of interest is our computational effort was quite lower than another algorithms. Ongoing investigation will consist in adaptation of algorithm to operate with multiproduct version and the inclusion of structural policies in order to improve the quality of service.

References

[1] Adulyasak, Y., J.-F. Cordeau and R. Jans, *Formulations and Branch-and-Cut Algorithms for Multivehicle Production and Inventory Routing Problems*, INFORMS Journal on Computing **26** (2014), pp. 103–120.

[2] Coelho, L. C., J.-F. Cordeau and G. Laporte, *Consistency in multi-vehicle inventory-routing*, Transportation Research Part C: Emerging Technologies **24** (2012), pp. 270–287.

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

[4] Desaulniers, G., J. r. G. Rakke and L. C. Coelho, *A branch-price-and-cut algorithm for the inventory-routing problem.*, Technical report, GERAD - Department of Mathematics and Industrial Engineering Polytechnique Montréal, Montreal (2014).

[5] Penna, P. H. V., A. Subramanian and L. S. Ochi, *An Iterated Local Search heuristic for the Heterogeneous Fleet Vehicle Routing Problem*, Journal of Heuristics **19** (2011), pp. 201–232.