

To appear in the *International Journal of Production Research*  
Vol. 00, No. 00, 00 Month 20XX, 1–15

## Robustness of inventory replenishment policies and client selection methods for the stochastic and dynamic inventory-routing problem

Raúl F. Roldán<sup>a\*</sup>, Rosa Basagoiti<sup>a</sup>, Leandro C. Coelho<sup>b</sup>

<sup>a</sup>*Electronics and Computing Department, Mondragon University, Goiru Kalea 2, Arrasate, Spain;*

<sup>b</sup>*Interuniveristy Research Centre on Enterprise Networks, Logistics and Transportation (CIRRELT) and Operations and Faculté des sciences de l'administration, Université Laval, Québec, Canada*

receivedv4.1 released January 2014

(Received 00 Month 20XX; final version received 00 Month 20XX)

When inventory management, distribution and routing decisions are made jointly implementing a vendor-managed inventory strategy, a difficult combinatorial optimization problem must be solved to determine how much to replenish, which customers to visit, and how to route the vehicles around them. We analyze a distribution system with one warehouse, one vehicle and many customers under the most commonly used inventory policy, namely the  $(s, S)$ , for different values of  $s$ , and three different customer selection methods: big orders first, lowest storage capacity first, and rest equal quantity. When customer demands are revealed gradually, ideal solutions are considered for benchmarking, using one step ahead advanced information. The system was analyzed using instances of different sizes regarding the number of vendors involved. The resulting vehicle route represents a solution for a traveling salesman problem. We compare the solutions of our genetic algorithms against those of Concorde and Lin-Kernigan methods to test the the robustness of the replenishment policies and client selection methods.

**Keywords:** Inventory-Routing Problem; Inventory Policies; Genetic algorithmim; Stochastic and Dynamic IRP; Demand management.

### 1. Introduction

Supply chain performance, coordination and integration are some key success factors in obtaining competitive advantages Moin and Salhi (2007). Inventory and distribution management are two main activities towards that integration, and are said to account for more than 60% of the total logistics costs Guasch (2008). The integration of inventory and distribution decisions gives rise to the inventory-routing problem (IRP), which has been studied for the past few decades and has received much attention lately Coelho, Cordeau, and Laporte (2014). However, most of these studies focus on optimizing a problem for which all information is known a priori, which is often not the case in practice.

The demand information in an IRP can be static when customers demand are known before the planning, or in a dynamic context in which it is gradually revealed over time (Psaraftis 1995; Coelho, Cordeau, and Laporte 2014). The dynamic and stochastic IRP (DSIRP) aims not at providing a static output, but rather a solution strategy that uses the revealed information, specifying which actions must be taken as time goes by (Bereglia, Cordeau, and Laporte 2010; Bertazzi et al. 2013). Recently, Bertazzi et al. (2013); Solyali, Cordeau, and Laporte (2012) and Coelho, Cordeau, and Laporte (2012a) have solved DSIRPs with the goal of minimizing the total inventory, distribution and shortage cost. They considered at least one of the classical inventory policies, i.e., maximum level or order-up-to. They tested their algorithms on instances containing several customers and periods.

---

\*Corresponding author. Email: raul.roltan@alumni.mondragon.edu, rbasagoiti@mondragon.edu, leandro.coelho@cirrelt.ca

An overview of state of the art of IRPs is provided in Roldán, Basagoiti, and Onieva (2014) where some key elements were identified that should be taken into account to propose alternative solutions to DSIRPs. The information management between different stakeholders in the supply chain is one of them, since this determines the evolution and quality of shared information. It is necessary to establish inventory management policies, which requires the information sharing between stakeholders. Inventory policies and their relation to the information on the demand is another one, in order to properly manage inventory levels. Finally, one must decide which optimization technique to use in order to make the best use of the available data.

The choice of which inventory policy to apply will largely influence the cost of the optimization process. Typically, it uses three parameters that can be related to the key questions to answer in inventory control: when replenish, how much to replenish, and how often the inventory level is reviewed. For periodic reviewed inventory, Wensing (2011) highlights three policies. One is the order-up-to (OU) which refers to a  $(t, S)$  system. Here, in each period  $t$ , the quantity delivered is that to fill its inventory capacity. Other policies are the  $(t, s, S)$  and the  $(t, s, q)$ . In the former, the customer is served if the inventory level is less than  $s$ . In the latter, the replenishment level  $q$  is flexible but bounded by the storage capacity available at each customer. The policies should be articulated with strategies for clients selection, because sometimes it is not possible to serve all clients, and in such cases, it is necessary to prioritize some of them.

Several exact, heuristic and metaheuristic methods have been used to find feasible solutions for such a problem and its variants, such as the vehicle routing, where branch-and-cut and evolutionary algorithms are widely used. Simic and Simic (2013) argued that for complex optimization problems such as the IRP, hybrid methods with techniques such as artificial neural networks, genetic algorithms, tabu search, simulated annealing and evolutionary algorithm can be successfully applied. Some techniques to solve IRPs are summarized in Table 1.

Following these ideas, in this paper we study a DSIRP in which decisions must be made without exact information about demand, which is gradually revealed over time. We propose a new three step solution algorithm, which is flexible enough to consider several inventory replenishment policies. We are then able to evaluate and compare the performance of the policies on demand satisfaction, average inventory kept at the customers' site, and total cost. Moreover, we show the effect of integrating tactical and operational decisions into the same solution algorithm. A sub-product of this research is the development of fast and efficient genetic algorithm to solve the traveling salesman problem appearing as subproblems from our solution methodology. This algorithm is compared directly against Concorde and the Lin-Kernigan heuristic in terms of solution quality.

The remainder of this paper is organized as follows. In Section 2 we formally describe the problem, and model it mathematically in Section 3. In Section 4 we present our solution procedure which includes customer selection, quantities determination, and vehicle routing. In Section 5, we present the results of extensive computational experiments and we analyze the trade-off between inventory and transportation costs. We describe how we can identify dominated solutions under a multi-objective optimization approach. In Section 6 we present our conclusions and findings.

## 2. Problem description

The IRP under study consists of one supplier and several vendors as depicted in Figure 1. We assume that the supplier has enough inventory to satisfy the demand of its customers. Customers demand are gradually revealed over time, thus it is said to be dynamic and unknown to the decision maker at the time all the decisions are made. The problem is defined over several periods, typically days, and without loss of generality we assume the demand becomes known at the end of the period. This demand can encompass a set of products organized in, e.g., a pallet, and we will then treat a single commodity as it is done in other IRPs. The supplier has a single capacitated vehicle to distribute the products and to satisfy the final demand of the customers.

PLEASE EDIT TO FIGURE TO: Supplier, Retailers, Final customers, Information flow, Product flow

A solution to this problem should determine the periods in which each customer must be visited, how

Table 1.: Metaheuristics used for IRP

<b>Technical</b>	<b>Use</b>	<b>Who</b>	<b>Description</b>
<b>Artificial Neural Networks</b>	Demand forecasting Price Forecasting To classify units of inventory	Jaipuria and Mahapatra (2014), Partovi and Anandaraajan (2001)	Learning technique to predict the behavior of a variable interest
<b>Genetic Algorithms</b>	To search for good parameters for a function or heuristic. Clustering retails to replenish by each of the vehicles available. To search optimal routes for replenishment retails.	Christiansen et al. (2011), Cheng and Wang (2009), Liu and Lee (2011)	To find good solutions in large search spaces
<b>Local Search</b>	Replenishment policy for inserts and removal new replenishment point into a retailer's schedule. Adjust the quantity to delivery to retailer's. Exclusive operators for solve special cases.	Qin et al. (2014), Javid and Azad (2010)	Neighborhood search strategies is applied to improve the initial solution. To avoid search cycling by systematically preventing moves that generate the solutions previously visited in the solution space.
<b>Simulated Annealing</b>	To improve an initial solution obtained from other heuristics and metaheuristics	Li et al. (2013), Liu and Lin (2005), Sajjadi and Cheraghi (2011)	To avoid premature solutions which are not good enough
<b>Rollout Algorithms</b>	To determine a shipping strategy that minimizes the expected total cost	Bertazzi et al. (2013)	These algorithms start with a heuristic policy and try to improve on that policy using on-line learning and simulation
<b>Branch and Cut</b>	To solution multiple products and multiple vehicles in IRP instances. To determine for each discrete time instant the quantity to ship to each retailer and the vehicle route.	Coelho and Laporte (2013), Archetti et al. (2007)	These methods work by solving a sequence of linear programming relaxations of the integer programming problem.
<b>Large Neighborhood Search</b>	To find good or near-optimal solutions by repeatedly trying to improve the current solution by looking for a better solution which is in the neighbourhood of the current solution	Coelho, Cordeau, and Laporte (2012b)	They present a formulation that allows transshipments, either from the supplier to customers or between customers.

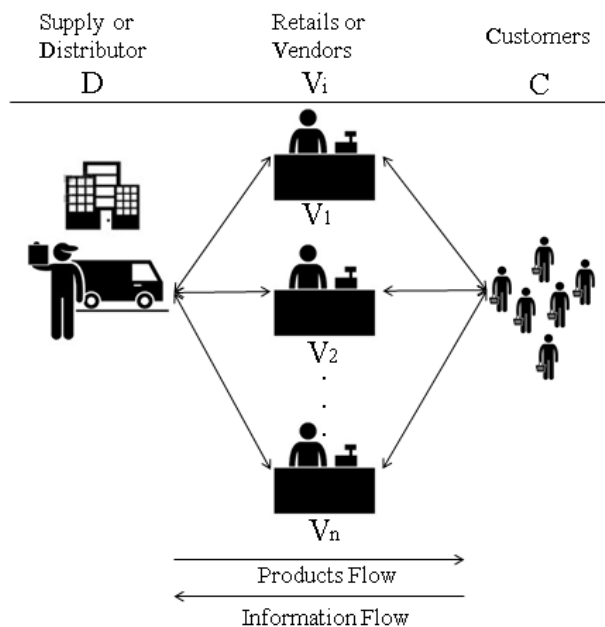


Figure 1.: A simplified IRP with one supplier,  $n$  retailers, and a set of final customers representing the demand of the retailers

much to deliver to each of them, and how to create vehicle routes that start at the supplier visit all customers selected to receive a delivery in the period, and return to the supplier location. All capacities must be respected, and stockouts are penalized in order to be avoided.

### 3. Mathematic formulation

In this section we present a mixed-integer linear programming formulation for the problem at hand. The costs incurred are the total of inventory and transportation costs. Inventory costs include the inventory holding and shortage penalties. A transportation cost is paid for each arc traversed by the vehicle. The transportation cost is based on a symmetric cost matrix. Let  $n$  represent the number of retailers, each with an initial inventory  $I_i^0$ , and with  $H_{ix50}$  representing the historical demand of the previous 50 time periods. The actual demand of customer  $i$  in period  $t$  is represented by  $D_i^t$ . Each customer has a maximum inventory capacity  $C_i$ , and a unit holding cost  $h_i$  is due. Shortages are penalized with  $z$  per unit. The planning horizon is  $P$  periods long. A single vehicle with capacity  $Q$  is available at the depot. The depot has an initial inventory  $I_0^0$ , and units in inventory incur a unit holding cost  $h_0$ . A symmetric transportation cost  $c_{ij}$  is known.

The variables used in our model are the following. Let  $I_0^t$  represent the inventory level at the depot in period  $t$  and  $I_i^t$  be the inventory level at retailer  $i$  at the end of period  $t$ . Let  $Q_i^t$  be the quantity of product delivered in period  $t$  to retailer  $i$ ,  $L_i^t$  be the number of units of product demanded by retailer  $i$  but not delivered at the end of period  $t$ , and  $X_{ij}$  be a binary variable indicating whether the arc going from node  $i$  to node  $j$  is used by the vehicle.

The **irp!** (**irp!**) is defined with a graph  $G = (N, A)$ , where  $N = \{0, \dots, n\}$  is the node set and  $A = \{(i, j) : i, j \in N, i \neq j\}$  is the arc set. Node 0 is the distributor and the remainder ones represent vendors and are denoted by  $V = \{1, \dots, n\}$ . The problem is defined over a finite time horizon  $H$  where each element is denoted as  $t \in P = \{1, \dots, p\}$ , and corresponds to one day.

At the end of each period  $t$ , the inventory level  $I_i$  for each vendor  $i$  is actualized. It is calculated based on: a) the demand of the vendor  $V_i$  ( $d_i^t$ ) in period  $t$ , b) The lost demand  $L_i^t$  in period  $t$ , c) the inventory level at previous period  $I_i^{t-1}$  and d) the quantity of inventory  $q_i^t$  that arrives in period  $t$ . Equation 1 explain how these values are used to recalculate the inventory levels.

For replenishment decisions, at the beginning of the each period  $t$ , the decision maker knows the inventory level  $I_i$  for each vendor  $V_i$  of the previous period  $t - 1$ . This variable is used to determine the quantity of

inventory  $q_i^t$  that will be shipped to each vendor  $V_i$  in the period  $t$  according with the policy of inventory that is used.

A unit inventory holding cost  $h_1$  and  $h_i$  is considered by the distributor and by vendor  $V_i$ , respectively, at each time period  $t$ . If the demand of vendor  $V_i$  is higher than his inventory level,  $I_i$ , a certain demand for the product is unfulfilled,  $L_i$ , and a shortage penalty  $z_i$ , proportional to the number of unsold product units is considered. As a consequence, the inventory costs are recalculated for each time period according to this rule.

$$I_i^t = I_i^{t-1} + q_i^t - d_i^t + L_i^t \quad \forall i = 1, \dots, n \tag{1}$$

where  $h_i I_i^t$  represents inventory holding cost at each vendor and distributor by time period and  $z_i L_i^t$  represents the cost of penalty by stock-outs by time period.

The problem has two constraints that are the inventory holding capacity of each vendor  $V_i$  denoted as  $C_i$  and the capacity of one only vehicle  $Q$  available for shipments. It is further assumed that the distributor has enough inventory to meet all the demand during the planning horizon  $H$ .

The distribution process is based on a version of the **vrp!** (**vrp!**), the Capacitated VRP **cvrp!** (**cvrp!**). In the **cvrp!**, the vehicle has a predefined capacity and it is related to a single central distributor. The vehicle is able to perform one route per period, from the distributor to a subset of the vendors. One transportation cost  $c_{ij}$  and one binary variable is associated by each arc  $(i, j) \in A$  where each arc vendor is assigned a binary  $x$  variable representing whether the arc has been chosen or not for replenishing, according to an inventory policy. The objective is to minimize the total cost (i.e., a weighted function of the number of routes and their length) to serve all the vendors. To calculate the transportation cost, it is assumed that the distance between every two nodes is symmetric, that is, the distance from  $i$  to  $j$  is the same as  $j$  to  $i$ .

The integer programming formulation for the problem is as follows:

Objective Function:

min

$$Cost^t = \sum_{t=1}^p \sum_{i=1}^{n+1} h_i I_i^t + \sum_{t=1}^p \sum_{i=2}^{n+1} z_i L_i^t + \sum_{t=1}^p \sum_{i=1}^{n+1} \sum_{j=1}^{n+1} c_{ij} X_{ij} \quad \forall i < j \tag{2}$$

Subject to:

$$\sum_{i=1}^n q_i^t \leq Q \quad i \in V \tag{3}$$

$$I_i^t \leq C_i \quad i \in N \tag{4}$$

$$\sum_{i=1}^{n+1} \sum_{k=1}^{n+1} x_{ik} + \sum_{j=1}^{n+1} \sum_{k=1}^{n+1} x_{kj} = 2 \quad k \in N \wedge i < k \wedge j > k \tag{5}$$

$$\sum_{i,j \in S} x_{ij} \leq S - 1 \quad (S \subset V, 3 \leq S \leq n - 3) \tag{6}$$

$$X_{ij} \in \{0, 1\} \tag{7}$$

The objective function in equation 2 is to minimize the sum of the inventory and transportation costs. Equation 3 and 4 are inventory constraints, where 3 is the vehicle capacity constraint and 4 is the maximum allowable inventory level constraint. Equation 5 to Equation 7 are transportation constraints, where 5 is a degree constraint, 6 is a subtour elimination constraints and 7 is a binary variable. This formulation associates a binary variable  $x_{ij}$  with each edge  $(i, j)$ , equal to 1 if and only if the edge appears along the tour.

### 4. Solution Procedure

To obtain the solution to the problem and in order to reduce the complexity, it is possible to decompose the **irp!** in different sub-problems. First is to select the best route to go from the distributor to a set of vendors. This problem is known as the **tsp!** (**tsp!**), a classical combinatorial optimization problem. **tsp!** details may be consulted in Rajesh Matai and Mittal (2010), For this work, we used 3 different variations of a genetic algorithm **ga!**. Also the best solutions found so far by means of using the heuristic Lin-Kernighan and the exact solution provided by Concorde were computed Applegate et al. (2003) and Applegate et al. (2006). Concorde is the cutting-plane-based exact TSP solver (using the QSOpt LP solver) and Lin-Kernighan is an implementation of the Chained-Lin-Kernighan heuristic for the TSP. To transport the products, some restrictions were added to the routes to be fulfilled by the vehicles, here is when this problem becomes a **vrp!**. The **vrp!**, variants and features can be consulted in Toth and Vigo (2001). Additionally, when the levels of consumption of clients and suppliers in order to maintain a continuous replacement are considered, an **irp!** is generated. Using different inventory policies to replenish the vendors it is possible for the **irp!** fulfill the three aims: i) establish the optimal inventory levels, ii) plan the volume and number of shipments and iii) ensure that deliveries suit the requirements of each product.

According with this, our process to obtain a solution for any **irp!** instance has four steps in this approach. In Figure 2, a simplified scheme of this process is shown.

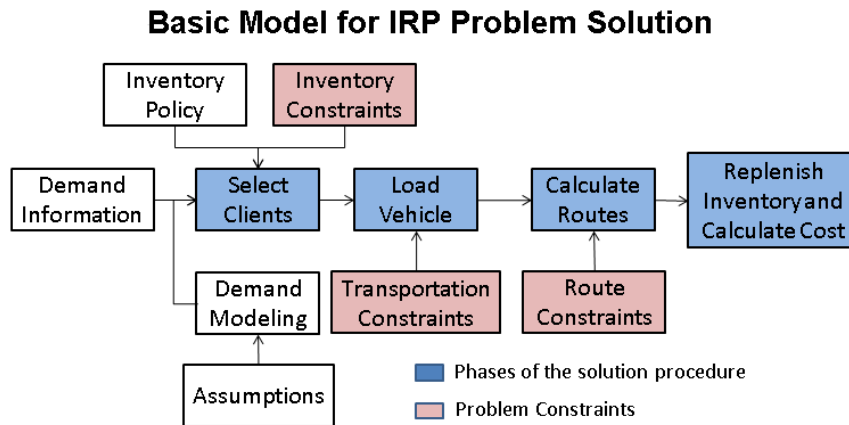


Figure 2.: Solution procedure

In the first phase, the vendors who are going to be replenished are selected using an inventory policy. In a second step the vehicle is loaded with the  $q_i$  units of product to be shipped to the vendors. The vendors to be attended are prioritized in case that the total load to be transported exceeds the capacity of the vehicle ( $Q$ ). This vendor selection process is performed for every time period  $t$  in the horizon time  $P$ . In a third phase, the route to be used to visit vendors is calculated. Finally, in the fourth phase, the total costs involved in the operation are calculated.

#### 4.1 Selecting vendors to replenish

The inventory policy guides the optimization process. In order to show the impact of each policy in the cost reduction of the system, different strategies are considered related to the cost of doing nothing, (not to replenish the vendors despite of demand). Comparing the results of each strategy to the cost of doing nothing give us the opportunity to obtain the level of reduction on the inventory costs because the transportation costs will be null.

7 different inventory policies were tested and details are shown in the Table 2, These inventory policies can be grouped together in five main different subgroups with 9, 1, 1, 7 and 1 policies respectively. All these policies are implemented only with slight modifications that arise in case the vehicle capacity is exceeded for the last vendor to be attended. An additional formulations of the problem have been solved in which

one day advanced demand information have been used to set the value of  $q_i$  in fixed quantity policy and the value of  $s$  in the  $(s, S)$  policy. These two solutions were used for benchmarking. For these two options, the values will be denoted in the same way as it was denoted the demand for a given period:  $d_i^t$ , the difference is that the value is known at the beginning of the day instead to be known at the end. In Table 2 all these policies are summarized.

Table 2.: Policies used to serve the vendors

Group	Variable	Policy	Decision
1	$q_{it} = \theta S$	<b>ml!</b> ( <b>ml!</b> ) and <b>ou!</b> ( <b>ou!</b> )	ML, if $I_i^t + q_{it} < C_i^t$ <b>ou!</b> , otherwise
2	$q_{it} = C_i^t - I_i^t$	<b>ou!</b>	<b>ou!</b> , if $C_i^t - I_i^t > 0$ 0, otherwise
3	$q_{it} = D_i^t$	<b>ml!</b> and <b>ou!</b>	ML, if $I_i^t + Q_i < C_i^t$ <b>ou!</b> , otherwise
4	$s = \alpha S$ where $\alpha = 0.25, 0.50, 0.75$ $s = \mu\{H_i\}$ $s = \mu\{H_i\} + \sigma\{H_i\} * z_\beta$ where $\beta = 0.05, 0.02, 0.01$	$(s, S)$ $(s, S)$ $(s, S)$	$q_{it} = S - I_i^t$ , if $I_i^t < s$ 0, otherwise
5	$s = D_i^t$	$(s, S)$	$q_{it} = S - I_i^t$ , if $I_i^t < s$ 0, otherwise

A deeper explanation of the policies at each group can be found next:

- (1) The decision maker chooses that the distributor ships always a **fq!** (**fq!**). This policy is called **ml!**, and the distributor allows to freely choose the quantity to deliver to the vendors, limited only by the inventory capacity of them. All vendors are replenished with a **fq!** by each period of time  $t$ . The **fq!** are defined for testing as fractions of maximum allowable inventory level for vendor as  $\theta S$  where  $\theta = \{0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1\}$ . It is important to note that in case of  $\theta = 0$ , nothing is shipped and in case of  $\theta = 1$  this policy becomes in the following policy **ou!**.
- (2) The decision maker chooses that the distributor ships a quantity  $q$  for each vendor depending on the difference between the maximum allowable and the current inventory level. This policy ensures that whenever a vendor is visited, the quantity delivered is that to fill its inventory capacity. In connection with the above policy, in case that the  $I_i$  inventory level plus  $q_i$  quantity fixed is greater than  $C_i$ , an **ou!** for each vendor is applied.
- (3) The decision maker knows one day advanced demand information, in this hypothetical case the  $q$  values is equal to  $D_i^t$ .
- (4) The decision maker chooses to implement an  $(s, S)$  inventory policy, which involves the parameters  $s$  and  $S$ . The value of  $S$  is assumed to be a fixed value that corresponds to the maximum allowable inventory level for each vendor, the parameter  $s$  is used to determine when to replenish. This  $(s, S)$  policy consists in ordering a variable quantity equal to the difference between a value  $S$  and the current inventory position  $I_i$  as soon as the inventory level is less than a value  $s$ . In order to test this policy, several values for  $s$  are considered and described below:
  - (a) The value of the parameter  $s$  is calculated for each vendor as one fraction of the maximum inventory capacity as  $s = \alpha S$  where  $\alpha = 0.25, 0.50, 0.75$ .
  - (b) The value of the parameter  $s$  is calculated for each vendor using the mean over its 50 historical data, so  $si = \mu\{H\}_i$
  - (c) The value of the parameter  $s$  is calculated for each vendor using the mean plus standard deviation of 50 historical data, so  $si = \mu\{H\}_i + z_\beta * \sigma\{H\}_i$  where  $\beta$  is the probability of a stock-out and  $z_\beta$  is the order quantile of the demand distribution.  $1 - \beta$  is usually referred to as the service level. Three different service levels of 95%, 98% and 99% were tested.
- (5) The value of the parameter  $s$  is equal to the one step ahead demand for the vendor  $D_i^t$ .

## 4.2 Loading the vehicle

For loading the vehicle, a version of the **vrp!** is used, the **cvrp!**. One vehicle is used with limited capacity and based on a single central distributor with multiple geographical dispersed vendors. The objective is to minimize the total transportation cost in each period  $t$  of the time horizon  $P$  to serve all the vendors.

Once the vendors have been chosen by means of the inventory policy, the vehicle is loaded to its maximum capacity  $Q$ . According to Toth and Vigo (2001), sometimes it is not possible to fully satisfy the demand of each customer. For these causes, a subset of the customers can be left unserved or the quantities to be shipped reduced. To deal with these situations, different priorities, or penalties associated with partial or total lack of service, can be assigned to the customers. In this work, three strategies were analyzed: a) prioritize vendors requiring more products **bof!** (**bof!**), b) Prioritize vendors with less storage capacity and three **lsf!** (**lsf!**) c) subtract the same amount to all orders until all vendors can be served **req!** (**req!**). For the first and second strategies, it is important to notice that the last vendor selected will only be replenished with the remaining capacity of the vehicle.

## 4.3 Calculating the near-optimal routes

In this subsection we explain the procedure for design the three **ga!**, whose variations will be in the selection process and details of Lin-Kernighan and Corcorde implementation.

A **ga!** (**ga!**) is used to calculate a near-optimal route visiting all the selected vendors to ship. The results are contrasted with the results obtained using the Concorde algorithm Applegate et al. (2006), Applegate et al. (2003) and Coelho, Cordeau, and Laporte (2012a). A cost for the routes is used as a fitness measure. The transportation costs have been determined by the euclidean distance between the nodes considered along the route.

For **ga!**, each gene of the chromosome is a number which identifies the distributor or the vendor. A gene with the number 1 identifies the distributor, others gene values identify the vendors and the order of the genes is the order in which the vendors will be served. The population, at the beginning, will be random.

The GA uses four biological operators: elite, selection, crossover and mutation. Using the operator of elite, the 20% percent of the individuals of the population with the best fitness pass directly into to new population. The remaining 80% pass through crossover and mutation operations.

Once some of the individuals were selected for crossover, with a probability of 50% percent, they will be using three different selection methods (**rw!** (**rw!**), **bt!** (**bt!**) and **bd!** (**bd!**)) to pick up the parents to be used for crossover.

A uniform crossover is used in order to generate two off-springs, each offspring has the 50% percent of characteristics of the father and the other 50% percent of the mother.

Finally, a mutation operator with probability of 50% is applied to the initial population were four different types of mutation were used: flip, insert, swap and scramble.

The final population will be formed by the following combination of operators: 20% by elite operator, 20% by selection operator, 20% by selection and mutation operators, 20% by selection and crossover operators and 20% by selection, crossover and mutation operations.

A simple stopping criterion is implemented: the algorithm stops if the best fitness for each generation remains constant a given number of generations. The number of generations are chosen to follow an exponential function on the number of vendors to attend. The output value is denoted by  $n_{cp}$  and if the stopping criteria is not fulfilled, the evolution process continues. Table 3 shows constant numbers determining the stopping criteria in relation with the complexity of the problem.

Table 3.: Simple stopping criterion

Instance	$n_{cp}$
< 5	1
< 10	20
< 50	70
< 100	200
Otherwise	550



#### 4.4 Inventory Replenishment and Total Costs Calculation

The total costs are calculated at the end of every period and the results of the variable of interest are obtained.

### 5. Outcomes

The instances used to test the system are denoted as follows: a name *irp* that mean "Inventory Routing Problem", *n* number of vendors, *p* number of periods and 1 the file number that specifies the specific instance used for this trials. For example, *irp* – 5 – 20 – 1, is a instance with 5 vendors and 20 periods and the specific file used for this trials is the one denoted by the number 1. The testing data, as previously stated, were the instances proposed by Coelho, Cordeau, and Laporte (2012a), in the specific case of standard instances.

Each method was tested for given solution to 10 different instances. Each instance was tested with 7 inventory policies and for each one of 3 strategies for select the clients. As a result of the process, 57 possible solutions were generated for each instance in order to compare the reductions in the cost in the **sc!** (**sc!**). These are the details about the resources used in this work. All computations were performed on a personal computer with Intel Core i3-2370M running Matlab R2009b on Windows 8.1 operating system. The processor running at 2,40GHz and with up to 8GB of RAM memory.

#### 5.1 The objective of reducing the total cost is analyzed

In order to minimize the total cost of the process of inventory and distribution, the two situations described below were analyzed. The first one used the groups 1, 2 and 3 of the inventory policies and the second one the remaining ones.

##### 5.1.1 The distributor sends a *q* quantity

In the hypothetical case that the decision maker chooses that the distributor does not replenish any vendors and "wait and see what happens", the total cost is the sum of inventory holding cost plus the cost by stockouts. The total cost for the system was summarized in Table 4. In this table, each instance for testing was classified in small, medium and large according to the number of vendors to attend. The total cost of the system were separated in inventory costs and transportation costs. Finally, in the last column, the stockouts costs were displayed.

Table 4.: Inventory and transportation cost in the case: "the distributor does not replenish any vendors"

Type Instance	Instance	Cost Inv	Cost Trans	Lost Ven
Small	IRP-5-20-1	45886,37	0,00	45588,00
Small	IRP-10-20-1	91507,07	0,00	91020,00
Small	IRP-15-20-1	154650,72	0,00	153808,00
Small	IRP-25-20-1	276301,55	0,00	275068,00
Medium	IRP-50-10-1	509891,8	0,00	506978
Medium	IRP-75-10-1	826527,12	0,00	822502
Medium	IRP-100-20-1	1174447,09	0,00	1169380
Large	IRP-125-20-1	1356658,42	0,00	1349788
Large	IRP-150-20-1	1616019,46	0,00	1608706
Large	IRP-200-20-1	2161450,02	0,00	2151808

The reduction of total costs begins when the decision maker chooses that the distributor replenishing the vendors by means of **fq!** by period, then, several  $\theta$  values in policie **fq!**, were tested to analyze the cost reduction. For every vendor, there is a point in the cost curve over time in which to send higher quantities than a certain value has no any effect. That is due that each vendor has limited inventory capacity, so the condition is to replenish  $q_i = \theta S$  if  $I_i^t + \theta S < C_i$  or otherwise,  $q_i = C - I_i^t$ .

In case that decision maker chooses that the distributor replenishes the vendors by means of variable quantities  $q_i$ . As each *t* period the inventory is reviewed, at vendor  $V_i$  is sending a  $q_i$  quantity equal to

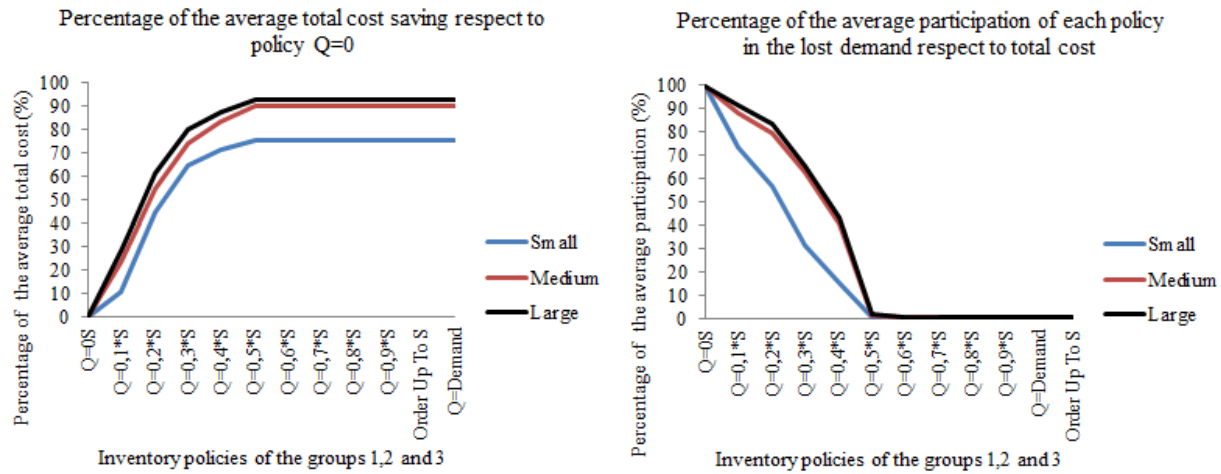


Figure 3.: Total cost saving respect to  $q = 0$  and participation of the lost demand respect to total cost for inventory policies in the groups 1, 2 and 3 for small, medium and large instances

difference between the maximum inventory allowed and its current inventory level. This policy is **ou!** and in this case is related with the maximum inventory level allowed.

In the hypothetical case that decision maker knows one day **ad!** (**ad!**) information, he decides replenish each vendor  $V_i$  with the real value of the demand  $q_i^t = D_i^t$  per period  $t$ .

The advantage of testing several options is that a curve of cost can be drawn and a tendency could be observed in the results. Thus, the costs were reduced until certain value of  $q_i$  was reached and next, the costs began to stabilize. The policies were compared and the trend of each method in reducing the costs displayed. Two graphics were obtained. Figure, 3 shows the total cost saving respect to  $q = 0$  on the left part, where from  $q = 0.5S$  to send higher quantities has no effect. In the same sense, the right part of the figure, shows the participation of the lost demand respect to the total cost, where from  $q = 0.6$  no losses are generated.

### 5.1.2 The distributor sends products according to the $(s, S)$ policy

With the assumption that the parameter  $S$ , the size of the maximum inventory level per vendor  $V_i$  is known and fixed to a predefined value, the parameter  $s$  was used for optimization and three different policies used to configure this parameter. The first option defines the value  $s$  as quarter, half and three quarter parts of the value of  $S$ , reducing this way the probably of stock-out to have safety stock. As second option, and using historical data, the value  $s$  is defined by means of the mean of 50 historical demand data. Finally, the parameter  $s$  was configured according to a service level. Thus, the standard deviation is added to the mean according to the desired service level. In this chapter, three service levels were proposed and these are: 95%, 98% and 99%. These service levels are compared with the previous policies.

In the hypothetical case that decision maker knows one day advanced demand information, he decides replenish each vendor  $V_i$  configuring the  $s$  parameter with real value of the demand  $s = D_i^t$  per period  $t$ , so guarantee at least the demand by period in attend by units storage in inventory.

As before, two graphics were obtained. The Figure4 shows, on the left part, the percentages of saving cost for above of the 75% with respect to  $q = 0$ . Medium and large instances reported greater savings respect to small instances. The right part of the Figure 4 shows the participation of the lost demand respect to total cost where from the policy  $s = D_i^t$  very low or no losses are generated.

## 5.2 An approach to the moo! analysis

A multi-objective optimization aims at finding Pareto-optimal set or Pareto front consisting of several solutions balancing conflicting objectives. Thus, a multi-objective optimization problem deals with simultaneous optimization of two or more objectives which are conflicting. They are conflicting because improvement

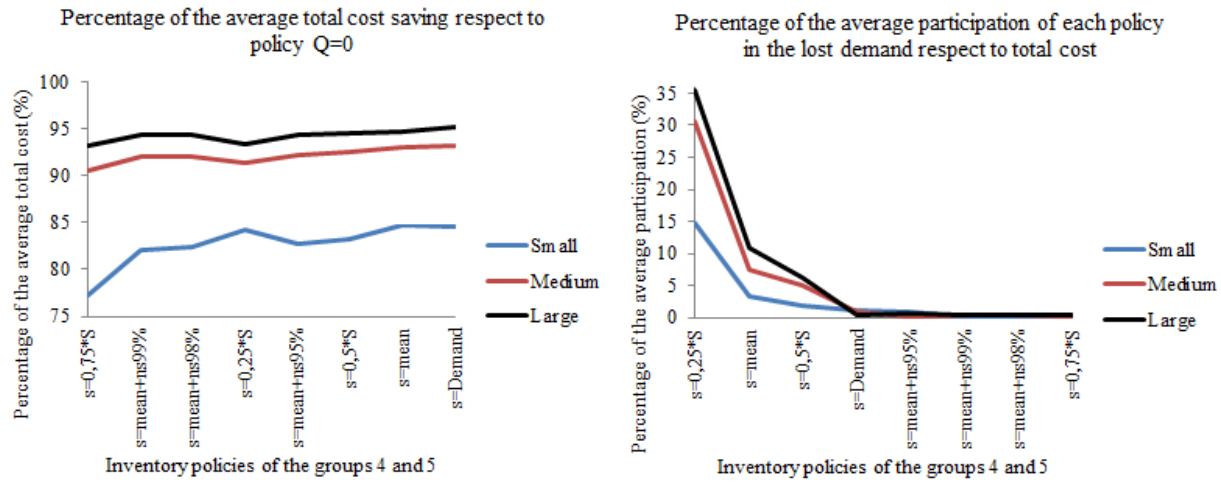


Figure 4.: Total cost saving respect to  $q = 0$  for different policies and participation of the lost demand respect to total cost for inventory policies in groups 4 and 5 for the small, medium and large instances

in any objective is not possible without degradation in other objectives.

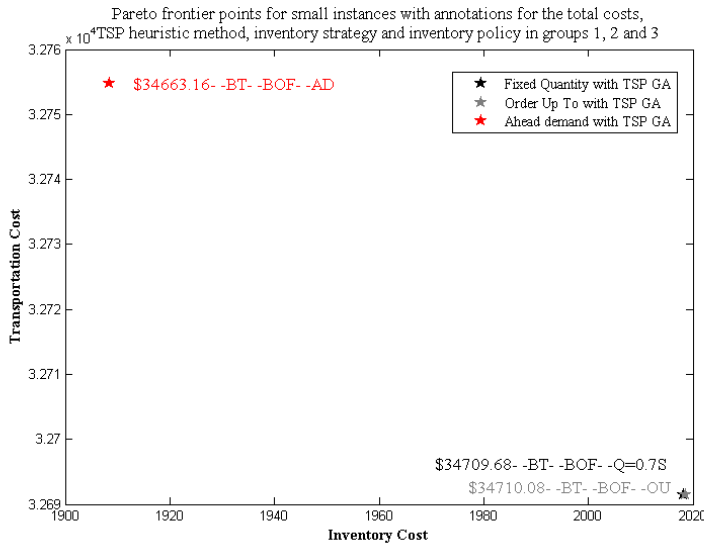


Figure 5.: Pareto frontier points for small instances with annotations for the total costs, **tsp!** heuristic method, inventory strategy and inventory policy in groups 1, 2 and 3

This is a case of the objective of the minimization of transportation cost and minimization of the inventory cost, hence there cannot be a single optimum solution which simultaneously optimizes all objectives. The resulting outcome of a **moo!** (**moo!**) is a set of optimal solutions with a varying degree of objective values. This set of solutions is called the non-dominated set or Pareto optimal set. Because minimization of transportation cost and minimization of the inventory cost cannot be achieved at the same time, there exists a trade-off between them. This type of system clearly represents a multi-objective optimization situation which is a procedure looking for a compromise policy, based on a number of options. Hence the Pareto set solutions and their corresponding decision variables should be provided, from which the decision-maker can select a solution to satisfy the industrial needs Shankar et al. (2013).

According with this, solutions found by the procedure were drawn as points in a plane, the Y axis representing the transportation cost and the X axis representing the inventory cost and the dominant solutions were searched. As before, the two situations described below were analyzed. The first one used the groups 1, 2 and 3 of the inventory policies and the second one the remaining. Each situation is analyzed for every

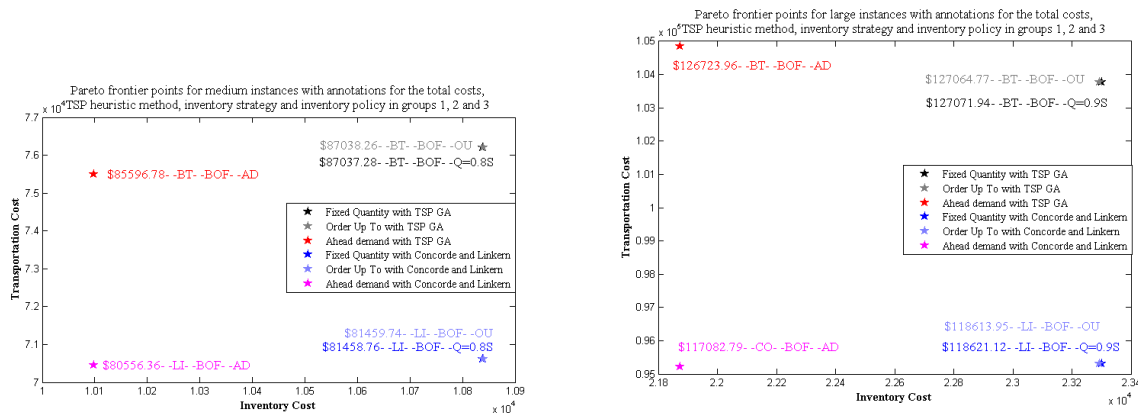


Figure 6.: Pareto frontier points for medium and large instances respectively with annotations for the total costs, **tsp!** heuristic method, inventory strategy and inventory policy in groups 1, 2 and 3

type of instances, thus 6 graphics are obtained.

Each figure represents the Pareto frontiers points with annotations for the total cost, TSP heuristic, inventory strategy and inventory policy. In these instances, medium and large, the results were compared with Linkern and Concord solution for the **tsp!**. Different types of dominant solutions are differentiated by color. The black color correspond with the solutions proposed in this paper.

In Figure 5, three types of dominant solutions **fq!**, **ou!** and **ad!** are found in the Pareto frontier. Although **ad!** solution is good in minimizing inventory costs, has a high cost in transportation. In contrast, **fq!** and **ou!** have a low cost in transportation but higher cost in inventory. It is important to note that these policies do not generate costs on lost demand.

In Figure 6, six types of dominant solutions are presented. The solutions obtained by **tsp!** can be improved in order to reduce the difference with the hypothetical case of **ad!**. Using Concorde and Linkern the solution are improved and show the same behavior. The results are influenced by the method of solutions. For transportation costs, the method of solution of the **tsp!** is what makes the difference.

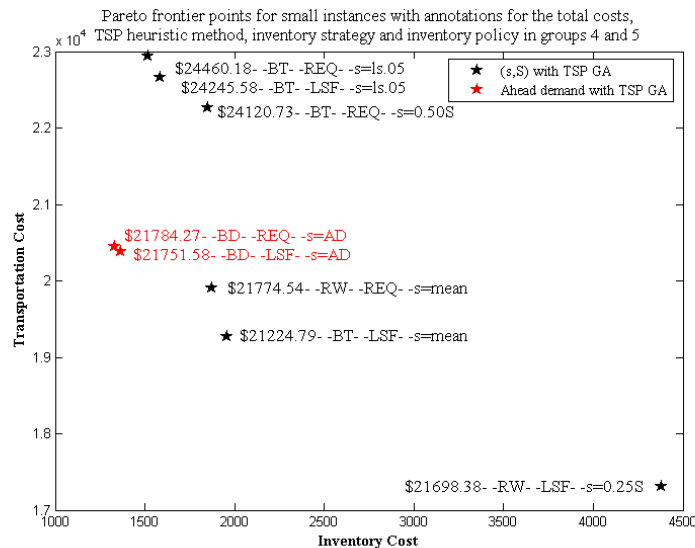


Figure 7.: Pareto frontier points for small instances with annotations for the total costs, **tsp!** heuristic method used, inventory strategy and inventory policy used for groups 4 and 5

In Figure 7, two types of dominant solutions ( $s, S$ ) policies and **ad!** for small instances are presented. Although the **ad!** solutions dominant some of the solutions; those located in the top of the figure, the solutions both **rw!** and **bt!** client selecting of the ( $s, S$ ) policy where  $s = mean$  and the ( $s, S$ ) policy where  $s =$

0.25S are dominant solutions. However these solutions generate higher lost demand cost than the policies (s, S) policy where  $s = 0.50S$  or policy (s, S) where  $s = \mu\{H_i\} + \sigma\{H_i\} * z_{0.05}$ .

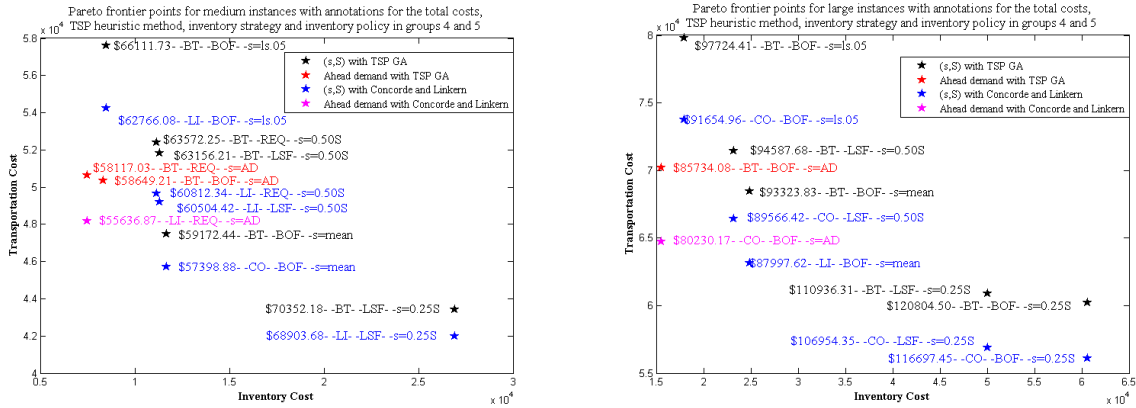


Figure 8.: Pareto frontier points for medium and large instances with annotations for the total costs, **tsp!** heuristic method, inventory strategy and inventory policy in groups 4 and 5

In Figure, 8 four types of dominant solutions (s, S) policies and **ad!** for medium and large instances are presented. As the previous figure, although the **ad!** solutions dominant some of the solutions, those located in the top part of the figure, the solutions both **rw!** and **bt!** of the (s, S) policy where  $s = mean$  and the (s, S) policy where  $s = 0.25S$ . However these solutions generate higher lost demand cost than the obtained with the policies (s, S) policy where  $s = 0.50S$  or policy (s, S) where  $s = \mu\{H_i\} + \sigma\{H_i\} * z_{0.05}$ . It is important to note that solution obtained by **tsp!** are near to the solutions obtained with Concorde and Linkern.

### 6. Conclusions

The **dsirp!** (**dsirp!**) brings the opportunity to use historical demand data forecasting to obtain better solutions, substantially improving the perform of the **sc!**, as the results obtained for the ideal situation simulated for the (s,S) strategy show. Forecasting procedures and probabilistic models could be used to predict and model the demand and improve the performance of the system, bringing it closer, to the best solution. This work was focused on testing this type of systems under uncertainty and compare genetic algorithm operators and strategies of clients selection methods. These test were extended to the Concorde and Linkern software, and also to solutions using advanced information, namely, ideal solutions. The results of the tests showed that there is room for improvement for using forecasting methods along all the instance types, regardless the complexity (small, medium or large instances, up to 200 vendors). Regarding the vendor selection methods explored, the first important point is that the strategy **bof!** is better than the others two strategies for the all instances in the policies **fq!**, **ou!** and **ad!**. In the other policies not only the **bof!**, but also **lsf!** is a good option. However, It is important to not that the strategy **req!** dominated in some solutions. In relation to the other strategy analyzed, related to the selection method for the genetic algorithm solving TSP, the binary tournament was most of the times the winner.

The model provided in this work analyzes **sc!** efficiency in terms of inventory and transportation costs according to certain inventory management strategies but is not a true bi-objective system. To get that, it will be necessary to implement a solution in which the representation for the heuristic method considers not only transportation but also inventory management options, considering higher levels of integration for inventory management and transportation. Furthermore, to increase flexibility in decision making, inventory costs and transportation costs criteria have to be analyzed simultaneously in an **moo!** approach.

## References

- Applegate, David, Robert Bixby, Vašek Chvátal, and William Cook. 2003. "Concorde: A code for solving traveling salesman problems." *World Wide Web*, <http://www.math.princeton.edu/tsp/concorde.html>.
- Applegate, David, Ribert Bixby, Vasek Chvatal, and William Cook. 2006. "Concorde TSP solver."
- Archetti, C.a, L.a Bertazzi, G.b Laporte, and M.G.a Speranza. 2007. "A branch-and-cut algorithm for a vendor-managed inventory-routing problem." *Transportation Science* 41 (3): 382–391. Cited By (since 1996)36.
- Berbeglia, Gerardo, Jean-François Cordeau, and Gilbert Laporte. 2010. "Dynamic pickup and delivery problems." *European Journal of Operational Research* 202 (1): 8 – 15. <http://www.sciencedirect.com/science/article/pii/S0377221709002999>.
- Bertazzi, L.a, A.b Bosco, F.c Guerriero, and D.d Laganá. 2013. "A stochastic inventory routing problem with stock-out." *Transportation Research Part C: Emerging Technologies* 27: 89–107.
- Cheng, Chi-Bin, and Keng-Pin Wang. 2009. "Solving a vehicle routing problem with time windows by a decomposition technique and a genetic algorithm." *Expert Systems with Applications* 36 (4): 7758 – 7763. <http://www.sciencedirect.com/science/article/pii/S0957417408006210>.
- Christiansen, M.a, K.b Fagerholt, T.c Flatberg, O.d Haugen, O.d Kloster, and E. H.e Lund. 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.a, and G.b Laporte. 2013. "A branch-and-cut algorithm for the multi-product multi-vehicle inventory-routing problem." *International Journal of Production Research*.
- Coelho, L. C., J.-F. Cordeau, and G. Laporte. 2012a. *Dynamic and Stochastic Inventory-Routing*. Tech. Rep. CIRRELT-2012-37. Montreal, Canada.
- Coelho, Leandro C., Jean-François Cordeau, and Gilbert Laporte. 2012b. "The inventory-routing problem with transshipment." *Computers & Operations Research* 39 (11): 2537 – 2548. <http://www.sciencedirect.com/science/article/pii/S0305054812000032>.
- Coelho, L. C., J.-F. Cordeau, and G. Laporte. 2014. "Thirty Years of Inventory-Routing." *Transportation Science* 48 (1): 1–19.
- Guasch, J.L. 2008. "Logistic Costs in Latin America and Caribbean."
- Jaipuria, Sanjita, and S.S. Mahapatra. 2014. "An improved demand forecasting method to reduce bullwhip effect in supply chains." *Expert Systems with Applications* 41 (5): 2395 – 2408. <http://www.sciencedirect.com/science/article/pii/S0957417413007926>.
- Javid, A. A.a, and N.b Azad. 2010. "Incorporating location, routing and inventory decisions in supply chain network design." *Transportation Research Part E: Logistics and Transportation Review* 46 (5): 582 – 597.
- Li, K.a, B.b Chen, A.I.c Sivakumar, and Y.d Wu. 2013. "An inventory-routing problem with the objective of travel time minimization." *European Journal of Operational Research*.
- Liu, S.C., and C.C. Lin. 2005. "A heuristic method for the combined location routing and inventory problem." *International Journal of Advanced Manufacturing Technology* 26 (4): 372–381. Cited By (since 1996)25. <http://www.scopus.com/inward/record.url?eid=2-s2.0-23944465435partnerID=40md5=c236332c1045e4146bdc69f6e11e01ea>.
- Liu, Shu-Chu, and Wei-Ting Lee. 2011. "A heuristic method for the inventory routing problem with time windows." *Expert Systems with Applications* 38 (10): 13223 – 13231.
- Moin, N.H.a, and S.b Salhi. 2007. "Inventory routing problems: A logistical overview." *Journal of the Operational Research Society* 58 (9): 1185–1194.
- Partovi, F.Y.a, and M.b Anandarajan. 2001. "Classifying inventory using an artificial neural network approach." *Computers and Industrial Engineering* 41 (4): 389–404. Cited By (since 1996)83. <http://www.scopus.com/inward/record.url?eid=2-s2.0-0036466985partnerID=40md5=776a0a4edcc06137c6c27ba526fe5718>.
- Psaraftis, H.N. 1995. "Dynamic vehicle routing: Status and prospects." *Annals of Operations Research* 61 (1): 143–164.
- Qin, L.a, L.b Miao, Q.c Ruan, and Y.d Zhang. 2014. "A local search method for periodic inventory routing

- problem.” *Expert Systems with Applications* 41 (2): 765–778.
- Rajesh Matai, Surya Singh, and Murari Lal Mittal. 2010. *Traveling Salesman Problem: an Overview of Applications, Formulations, and Solution Approaches, Traveling Salesman Problem, Theory and Applications*. Intech.
- Roldán, Raúl, Rosa Basagoiti, and Enrique Onieva. 2014. “Inventory Routing Problem with Stochastic Demand and Lead Time: State of the Art.” In *International Joint Conference SOCO14-CISIS14-ICEUTE14*, 73–82. Springer.
- Sajjadi, S.R.a, and S.H.b Cheraghi. 2011. “Multi-products location-routing problem integrated with inventory under stochastic demand.” *International Journal of Industrial and Systems Engineering* 7 (4): 454–476. Cited By (since 1996)9. <http://www.scopus.com/inward/record.url?eid=2-s2.0-79954617447partnerID=40md5=03c194db7b6dafadeaab12043c25d11e>.
- Shankar, B. Latha, S. Basavarajappa, Rajeshwar S. Kadadevaramath, and Jason C.H. Chen. 2013. “A bi-objective optimization of supply chain design and distribution operations using non-dominated sorting algorithm: A case study.” *Expert Systems with Applications* 40 (14): 5730 – 5739. <http://www.sciencedirect.com/science/article/pii/S0957417413002352>.
- Simic, D.a, and S.b Simic. 2013. “Evolutionary Approach in Inventory Routing Problem.” In *Advances in Computational Intelligence*, Vol. 7903 of *Lecture Notes in Computer Science* edited by Ignacio Rojas, Gonzalo Joya, and Joan Cabestany. 395–403. Springer Berlin Heidelberg. [http://dx.doi.org/10.1007/978-3-642-38682-4\\_2](http://dx.doi.org/10.1007/978-3-642-38682-4_2).
- Solyali, O., J.-F. Cordeau, and G. Laporte. 2012. “Robust inventory routing under demand uncertainty.” *Transportation Science* 46 (3): 327–340. Cited By 0. <http://www.scopus.com/inward/record.url?eid=2-s2.0-84866371094partnerID=40md5=b2f2060f80053e9fdd7a436eaf4ae275>.
- Toth, Paolo, and Daniele Vigo, eds . 2001. *The Vehicle Routing Problem*. Philadelphia, PA, USA: Society for Industrial and Applied Mathematics.
- Wensing, T. 2011. “Periodic Review Inventory Systems.” *Lecture Notes in Economics and Mathematical Systems* 651.