Journal of Heuristics, 11: 393–419, 2005

Minimizing the Total Cost in an Integrated Vendor—Managed Inventory System

LUCA BERTAZZI
Dip. di Metodi Quantitativi–Università di Brescia, Italy
email: bertazzi@eco.unibs.it

GIUSEPPE PALETTA
Dip. di Economia e Statistica–Università della Calabria, Italy

M. GRAZIA SPERANZA
Dip. di Metodi Quantitativi–Università di Brescia, Italy

Submitted in September 2003 and accepted by Jean-François Cordeau and Jean-Yves Potvin in February 2005 after 2 revisions

Abstract

In this paper we consider a complex production-distribution system, where a facility produces (or orders from an external supplier) several items which are distributed to a set of retailers by a fleet of vehicles. We consider Vendor-Managed Inventory (VMI) policies, in which the facility knows the inventory levels of the retailers and takes care of their replenishment policies. The production (or ordering) policy, the retailers replenishment policies and the transportation policy have to be determined so as to minimize the total system cost. The cost includes the fixed and variable production costs at the facility, the inventory costs at the facility and at the retailers and the transportation costs, that is the fixed costs of the vehicles and the traveling costs. We study two different types of VMI policies: The order-up-to level policy, in which the order-up-to level quantity is shipped to each retailer whenever served (i.e. the quantity delivered to each retailer is such that the maximum level of the inventory at the retailer is reached) and the fill-fill-dump policy, in which the order-up-to level quantity is shipped to all but the last retailer on each delivery route, while the quantity delivered to the last retailer is the minimum between the order-up-to level quantity and the residual transportation capacity of the vehicle. We propose two different decompositions of the problem and optimal or heuristic procedures for the solution of the subproblems. We show that, for reasonable initial values of the variables, the order in which the subproblems are solved does not influence the final solution. We will first solve the distribution subproblem and then the production subproblem. The computational results show that the fill-fill-dump policy reduces the average cost with respect to the order-up-to level policy and that one of the decompositions is more effective. Moreover, we compare the VMI policies with the more traditional Retailer-Managed Inventory (RMI) policy and show that the VMI policies significantly reduce the average cost with respect to the RMI policy.

Key Words: Vendor-Managed Inventory VMI, transportation, inventory, heuristics, logistics

Introduction

The integration of production, distribution and inventory management is one of the challenges of today's competitive environment. In the last decade the importance of the relations between internal management and external environment has been widely recognized, and the expression “supply chain management”, which emphasizes the view of the company as part of the supply chain, has become of common use. Sometimes the expression “coordinated supply chain management” is used to emphasize the coordination among the different components of the supply chain. The availability of data and information tools which derive from the advances in technology and communication systems has created the conditions for the coordination inside the supply chain. Different models for the integration of decisions of the production, inventory and distribution functions have been proposed. We refer to Cohen and Lee (1988) for a strategic stochastic model for a complete supply chain, in which four different functions are considered (material control, production control, finished good stockpile and distribution network control). A hierarchical decomposition approach is proposed to solve the problem. Thomas and Griffin (1996) review the coordination issue of the three functional areas at an operational level when deterministic models are used. Blumenfeld et al. (1985) analyze the trade-offs between transportation, inventory and production set-up costs over an infinite time horizon. Different shipping policies (direct shipping, shipping through a consolidation terminal and a combination of them) are studied on the basis of several simplifying assumptions. Sarmiento and Nagi (1999) and Erengüç, Simpson, and Vakharia (1999) review the integration between production and transportation. Chandra and Fisher (1994) propose a computational study to evaluate the value of coordination between production and distribution planning over a finite time horizon. Hall and Potts (2003) study the problem of minimizing the scheduling and delivering costs in the supply chain.

The availability of new information technologies has also led in the last years to the development of new forms of relationships in the supply chain. One of these is the so called Vendor-Managed Inventory (VMI), in which the supplier monitors the inventory of each retailer and decides the replenishment policy of each retailer. The supplier is responsible of the inventory level of each retailer and acts as a central decision-maker; therefore, she/he has to solve an integrated inventory-routing or production-inventory-routing problem. The advantage of the application of a VMI policy with respect to the traditional retailer-managed inventory policies relies in a more efficient utilization of the resources: The supplier can reduce its level of inventories maintaining the same level of service or can increase the level of service and can reduce the transportation cost by a more uniform utilization of the transportation capacity. On the other hand, the retailers can devote fewer resources to monitor their inventories and to place orders, having thus the guarantee that no stock-out will occur. An introduction to both deterministic and stochastic inventory routing problems can be found in Campbell et al. (1998). A decomposition approach has been applied to the solution of a complex inventory routing problem by Campbell and Savelsbergh (2004). A delivery schedule is created first, followed by the construction of a set of delivery routes. A strategic stochastic model has been presented by Webb and Larson (1995) with the objective of determining the optimal fleet size. We refer to Axsäter (2001), Çetinkaya and Lee (2000), Cheung and Lee (2002), Fry, Kapuscinski, and Olsen (2001), Kleywegt, Nori, and Savelsbergh (2002, 2004) and Rabah and Mahmassani (2002) for applications of the VMI policy to systems with stochastic demand.

We study an integrated model in which several items are produced at a production facility and shipped to several retailers over a finite time horizon by applying a VMI policy.

Shipments from the production facility to the retailers are performed by a fleet of vehicles. Each vehicle has a given transportation capacity. The total transportation cost is given by the sum of a fixed cost and the routing cost. The fixed transportation cost is charged for each vehicle used at least once during the time horizon. When the fixed transportation cost is set to 0, this model can also handle the case of outsourced transportation and all the situations where the fixed cost of vehicles is not relevant. Each item is absorbed by the retailers in a deterministic and time-varying way. Each retailer determines a maximum and a minimum level of the inventory of the items and can be visited several times during the time horizon. Every time a retailer is visited, the quantity delivered is such that the maximum level of inventory is reached, i.e. the order-up-to level quantity is shipped to the retailer. This inventory policy is inspired, in a deterministic setting, by the classical order-up-to level policy, widely studied in inventory theory. We refer to Axsäter and Rosling (1994) for an overview of inventory policies in multi-level systems and to Bertazzi, Paletta, and Speranza (2002) for an application of the deterministic order-up-to level policy to an inventory-routing problem. The production facility monitors the inventory of each retailer and is responsible to maintain the right level of the inventory at each retailer, i.e. a VMI policy is applied. The problem is to determine the production policy and the shipping policy that minimize the total cost, given by the sum of fixed and variable production cost, fixed transportation cost, routing cost and inventory cost both at the production facility and at the retailers. The production policy consists of determining for each discrete time instant the quantity of each item to produce, while the shipping policy consists of determining for each delivery time instant the set of retailers to visit, the quantity of each item to ship to each visited retailer and the route that each vehicle has to travel. The approach we present in this paper is general enough to be easily extended to variants of this problem, for instance to the case where there is a limit on production per time period. Since data are usually uncertain in practice, their average values can be used. To account for the variability of data, in a real life context the model can be used with a rolling time horizon.

The scope of the paper is four-fold. First, we aim at solving the above problem. Since the problem is very complex and the exact solution would be impractical, we propose heuristic algorithms to solve it. In particular, we decompose the problem into two subproblems, one concerning the production and one concerning the distribution, on the basis of two different criteria. We show that, for reasonable initial values of the variables, the order in which the subproblems are solved does not influence the final solution. We will first solve the distribution subproblem and then the production subproblem. This gives rise to two different solution methods, one for each of the decomposition criteria considered. In each of these methods, the subproblem concerning the production is optimally solved, while the subproblem concerning the distribution is solved by applying a constructive heuristic algorithm in which at each iteration a retailer is inserted in the solution. For each retailer, the heuristic builds a network in order to represent the incremental cost due to the insertion of the retailer in the solution. A shortest path in the network identifies a "min cost" way for adding the retailer to the solution. The solution obtained by hierarchically solving the two subproblems is improved by applying two procedures which try to coordinate production and distribution. This improvement phase is a new feature of our decomposition approach, with respect to other decomposition approaches, and we will show it improves the final solution. Secondly, we aim at analyzing the impact of the different decompositions and of the different rankings on the solution. Thirdly, we partially relax the order-up-to level policy in order to obtain a different replenishment policy, referred to as fill-fill-dump, frequently applied in practice (see Berman and Larson, 2001, for an application of this policy to an inventory-routing problem with stochastic demand). In this policy, the order-up-to level quantity is shipped to all but the last retailer served on each delivery route, while the quantity delivered to the last retailer is the minimum between the order-up-to level quantity and the residual transportation capacity of the vehicle. Finally, we aim at comparing the application of the VMI policies with respect to a conventional Retailer-Managed Inventory (RMI) policy.

The paper is organized as follows. In Section 1 the problem is formally described. For the sake of simplicity, we focus on a single item; the extension to the multi-item case is straightforward. In Section 2, the two problem decompositions and the corresponding four order-up-to level algorithms are described. Then, theoretical results show the impact of the two different rankings on the solution and the exact and heuristic algorithms used to solve the subproblems are described. In Section 3 the fill-fill-dump policy is introduced and the corresponding solution algorithms described. Finally, in Section 4, the computational results obtained on randomly generated problem instances are shown and discussed.

1. Problem description

We consider a production-distribution system in which one item is produced at a production facility 0 and shipped to a set $M = \{1, 2, \dots, i, \dots, n\}$ of retailers over a set $T = \{1, 2, \dots, t, \dots, H\}$ of discrete time instants. A quantity r_{it} of the item is absorbed by each retailer $i \in M$ in each time $t \in T$. A starting level of the inventory is given at the production facility B_0 and at each retailer i (I_{i0}). Since the level of the inventory at the retailers at the end of the time horizon can be different from the starting one, the problem is not periodic. Each retailer i defines a maximum level U_i and a minimum level L_i of the inventory of the item. We assume, without loss of generality, that $L_i = 0$. If retailer i is visited at time t, then the quantity x_{it} shipped to retailer i at time t is such that the level of the inventory in i reaches its maximum level U_i (order-up-to level policy). A quantity y_t (possibly 0) is produced at the production facility 0 in each time instant $t \in T$. The total production cost is given by the sum of a fixed set-up cost K, which is charged for each time instant $t \in T$ with $y_t > 0$, and of a variable production cost p, which is charged for each unit produced during the time horizon (see for instance the classical paper by Wagner and Whitin, 1958). The total production cost can be formulated as follows:

$$\sum_{t \in T} (K\delta(y_t) + py_t), \tag{1}$$

where $\delta(y_t) = 1$ if $y_t > 0$, and 0 otherwise.

Shipments from the production facility to the retailers are performed by a fleet of vehicles. Each vehicle v has a given capacity C and routing is allowed. We assume that any subset of retailers can be visited in any time instant, each retailer cannot be visited by more than once in the each time instant and that $C \geq \max_i U_i$ to guarantee feasibility. The transportation cost c_{ij} from i to j, with $i, j \in M' = M \cup \{0\}$, is known. The total transportation cost is given by the sum of a fixed transportation cost f, which is charged if vehicle v is used at least once during the time horizon, and the routing cost. In particular, if R_{tv} is the route traveled by vehicle v at time t, $z_{ijtv} = 1$ if node $j \in M'$ is the successor of node $i \in M'$ in the route R_{tv} and 0 otherwise, and $\delta(v) = 1$ if $\cup_{t\in T} R_{tv} \neq \emptyset$ and 0 otherwise. Then the total transportation cost is:

$$\sum_{v\in V} f\delta(v) + \sum_{t\in T}\sum_{v\in V}\sum_{i\in M'}\sum_{j\in M'} c_{ij} z_{ijtv}, \tag{2}$$

where $V = \{1, 2, \ldots, v, \ldots, n\}$ is the set of vehicles. The cardinality of V is equal to the number of retailers n, as the maximum number of vehicles used in any feasible solution cannot be greater than n, i.e. the worst-case happens when all the retailers have to be simultaneously served by a full load direct shipping at least once in the time horizon. The inventory cost is charged both at the production facility and at each retailer. If we denote by h_i the unit inventory cost at node $i \in M'$, then the total inventory cost over the time horizon can be computed as follows. At the production facility the level of the inventory at time t is given by the level at time $t-1$, plus the quantity produced at time t, minus the total quantity shipped to the retailers at time t, that is

$$B_t = B_{t-1} + y_t - \sum_{i\in M} x_{it},$$

which can be also written as $B_t = B_0 + \sum_{j=1}^{t} y_t - \sum_{i\in M}\sum_{j=1}^{t} x_{it}$. Therefore, we assume, as in Wagner and Whitin (1958), that the quantity produced at time t can be shipped to the retailers at time t. Moreover, we assume, without loss of generality, that the starting level of the inventory B_0 is 0. Therefore, the total inventory cost at the production facility is:

$$\sum_{t\in T} h_0 B_t = \sum_{t\in T}\sum_{j=1}^{t} h_0 y_j - \sum_{i\in M}\sum_{t\in T}\sum_{j=1}^{t} h_0 x_{ij}. \tag{3}$$

At each retailer $i \in M$, the level of the inventory at time t is given by the level at time $t-1$, plus the quantity of item shipped from the production facility to the retailer i at time t, minus the quantity absorbed at time t, that is

$$I_{it} = I_{it-1} + x_{it} - r_{it},$$

which can be also written as $I_{it} = I_{i0} - \sum_{j=1}^{t} r_{ij} + \sum_{j=1}^{t} x_{ij}$. The starting level of the inventory I_{i0} is given. Therefore, the total inventory cost at the retailers is:

$$\sum_{i\in M}\sum_{t\in T} h_i I_{it} = \sum_{i\in M}\sum_{t\in T} h_i \left(I_{i0} - \sum_{j=1}^{t} r_{ij}\right) + \sum_{i\in M}\sum_{t\in T}\sum_{j=1}^{t} h_i x_{ij}. \tag{4}$$

The problem is to determine, for each retailer $i \in M$, a set $D_i \subseteq T$ of delivery time instants (and therefore the quantity x_{it} to ship to the retailer i at each delivery time instant $t \in D_i$) and, for each time instant $t \in T$, the quantity y_t to produce at the production facility and the set of routes R_{tv}, $v \in V$, that allow us to visit all the retailers served at time t. These decision variables have to satisfy the following constraints:

1. *Order-up-to level constraints*: These constraints guarantee that the quantity x_{it} shipped to each retailer i in each time instant t is such that the level of inventory in i reaches its maximum level U_i at time t after having received the quantity x_{it} and absorbed the quantity r_{it}, i.e. x_{it} is either equal to $U_i - I_{it-1} + r_{it}$ if a shipment to i is performed at time t or equal to 0 otherwise. They can be formulated as follows:

$$x_{it} = (U_i - I_{it-1} + r_{it})w_{it} \quad i \in M \quad t \in T \tag{5}$$

where w_{it} is equal to 1 if the retailer i is served at time t and 0 otherwise.
2. *Stock-out constraints at the production facility*: They guarantee that the level of the inventory B_t is non-negative in each time instant $t \in T$:

$$B_t \geq 0 \quad t \in T. \tag{6}$$

3. *Stock-out constraints at the retailers*: They guarantee that for each retailer $i \in M$ the level of the inventory I_{it} in each time instant $t \in T$ is non-negative:

$$I_{it} \geq 0 \quad i \in M \quad t \in T. \tag{7}$$

4. *Routing constraints*: They guarantee that, for each time instant $t \in T$, a feasible set of routes is determined to visit all the retailers served at time t, that is the total quantity loaded on each vehicle is not greater than the transportation capacity:

$$\sum_{i \in R_{tv}} x_{it} \leq C \quad v \in V \quad t \in T \tag{8}$$

and the routes followed by the vehicles are feasible (see Toth and Vigo, 2002).

The objective function is to minimize the total cost, given by the sum of the total production cost (1), the total transportation cost (2) and the total inventory cost at the production facility (3) and at the retailers (4).

2. Solution algorithms

The problem described in the previous section, referred to as Problem $\mathcal{P}$, is NP-hard, since it reduces to the VRP in the class of instances in which the time horizon is made by one time instant only, the fixed and variable production costs, the inventory costs and the fixed transportation cost are zero and all the retailers need to be served. A natural way to heuristically solve the problem is to decompose it into subproblems and then to hierarchically solve them. We propose two different decompositions of the problem.

2.1. Problem decompositions

A first decomposition can be obtained by separating production from distribution. Two subproblems are obtained. The first determines, for each time instant $t \in T$, the quantity y_t to produce at the production facility, given the values $\hat{x}_{it}$ of the variables x_{it}. We denote by $\hat{x}$ the vector of the given values of the x_{it}'s. The aim is to minimize the sum of the total production cost and of the inventory cost at the production facility. This subproblem can be formulated as follows.

Subproblem $P(\hat{x})$

$$\min \quad \sum_{t \in T} (K\delta(y_t) + py_t + h_0 B_t) \tag{9}$$

$$B_t = B_{t-1} + y_t - \sum_{i \in M} \hat{x}_{it} \quad t \in T \tag{10}$$

$$B_0 = 0 \tag{11}$$

$$B_t \geq 0 \quad t \in T \tag{12}$$

$$y_t \geq 0 \quad t \in T. \tag{13}$$

Note that the objective function (9) can be also written as $-\sum_{i \in M} \sum_{t \in T} \sum_{j=1}^{t} h_0 \hat{x}_{ij} + \min \sum_{t \in T} (K\delta(y_t) + py_t + \sum_{j=1}^{t} h_0 y_j)$ and that this subproblem is the classical uncapacitated dynamic lot size problem proposed by Wagner and Whitin (1958).

The second subproblem concerns the distribution only. It allows us to determine for each retailer i the set D_i of delivery time instants (and therefore the quantity x_{it} to deliver to each retailer i in each delivery time instant $t \in D_i$) and for each time instant t the set of routes that visit all the retailers served at time t. The aim is to minimize the sum of the inventory cost at the retailers and of the total transportation cost. This subproblem is solved for given values $\hat{y}_t$ of the variables y_t. We denote by $\hat{y}$ the vector of the given values of the y_t's. This subproblem can be formulated as follows.

Subproblem $D(\hat{y})$

$$\min \quad \sum_{i \in M} \sum_{t \in T} h_i I_{it} + \sum_{v \in V} f\delta(v) + \sum_{t \in T} \sum_{v \in V} \sum_{i \in M'} \sum_{j \in M'} c_{ij} z_{ijtv} \tag{14}$$

subject to: the order-up-to level constraints (15)

the stock-out constraints at the retailers (16)

the routing constraints (17)

$$B_t = B_{t-1} + \hat{y}_t - \sum_{i \in M} x_{it} \quad t \in T \tag{18}$$

$$B_0 = 0 \tag{19}$$

$$B_t \geq 0 \quad t \in T. \tag{20}$$

Note that the objective function (14) can be also written as $\sum_{i \in M} \sum_{t \in T} h_i (I_{i0} - \sum_{j=1}^{t} r_{ij}) + \min \sum_{i \in M} \sum_{t \in T} \sum_{j=1}^{t} h_i x_{ij} + \sum_{v \in V} f\delta(v) + \sum_{t \in T} \sum_{v \in V} \sum_{i \in M'} \sum_{j \in M'} c_{ij} z_{ijtv}$.

A second decomposition of the Problem $\mathcal{P}$ into subproblems is obtained by moving the variable production cost and the part of the inventory cost at the production facility which depends on the variables x's only from the objective function of the subproblem $P(\hat{x})$ to the objective function of the subproblem $D(\hat{y})$. The rationale is that these two cost components are constant in the former problem, while they affect the optimization in the latter. In fact, the variable production cost $\sum_t py_t$ can be expressed as $\sum_i \sum_t px_{it}$, as it is optimal to produce during the time horizon exactly the quantity shipped to the retailers during the time horizon. The part of the inventory cost of the production facility which depends on the x's only is $-\sum_i \sum_t \sum_{j=1}^{t} h_0 x_{ij}$ (see (3)). We define $P'(\hat{x})$ and $D'(\hat{y})$, respectively, the subproblems obtained by moving the two cost components from the objective function of the subproblem $P(\hat{x})$ to the objective function of the subproblem $D(\hat{y})$. The subproblem $P'(\hat{x})$ can be described as follows:

Subproblem $P'(\hat{x})$

$$\min \quad \sum_{t \in T} \left(K\delta(y_t) + \sum_{j=1}^{t} h_0 y_j \right) \tag{21}$$

subject to (10)–(13),

while the subproblem $D'(\hat{y})$ is:

Subproblem $D'(\hat{y})$

$$\min \quad \sum_{i \in M} \sum_{t \in T} h_i I_{it} + \sum_{v \in V} f\delta(v) + \sum_{t \in T} \sum_{v \in V} \sum_{i \in M'} \sum_{j \in M'} c_{ij} z_{ijtv} + \sum_{i \in M} \sum_{t \in T} px_{it} - \sum_{i \in M} \sum_{t \in T} \sum_{j=1}^{t} h_0 x_{ij} \tag{22}$$

subject to (15)–(20).

Note that the objective function (22) can be also written as

$\sum_{i \in M} \sum_{t \in T} h_i (I_{i0} - \sum_{j=1}^{t} r_{ij}) + \min \sum_{i \in M} \sum_{t \in T} \sum_{j=1}^{t} (h_i - h_0) x_{ij} + \sum_{v \in V} f\delta(v) + \sum_{t \in T} \sum_{v \in V} \sum_{i \in M'} \sum_{j \in M'} c_{ij} z_{ijtv} + \sum_{i \in M} \sum_{t \in T} px_{it}$.

2.2. *Hierarchical algorithms*

In this section we present two hierarchical algorithms to solve Problem $\mathcal{P}$ for each of the two above presented decompositions. The first, referred to as VMI-PDP, is similar to the one proposed by Chandra and Fisher (1994). In the VMI-PDP, first the production subproblem is solved assuming that all retailers are served every day, i.e. $\hat{x}_{it}$ is equal to the quantity that would be shipped to retailer i at time t if i will be served every time instant. Then, given the production quantities, the distribution subproblem is solved. Given the quantity to ship to each retailer in each time instant, the production subproblem is solved again.

Heuristic VMI-PDP

1. Solve the subproblem $P(\hat{x})$, with $\hat{x}_{i1} = U_i - I_{i0} + r_{i1}$ and $\hat{x}_{it} = r_{it}$, $t > 1$. Let $\hat{y}$ be the vector of the obtained values of y_t, $t \in T$.
2. Solve the subproblem $D(\hat{y})$. Let $\hat{x}$ be the vector of the obtained value of the x_{it}, $i \in M$, $t \in T$.
3. Solve the subproblem $P(\hat{x})$.

The second algorithm, referred to as heuristic VMI-DP, is based on first solving the distribution subproblem, fixing as initial production quantity $\hat{y}_t$ at time $t \in T$ the quantity sufficient to serve the retailers $i \in M$ in each time instant $t \in T$, and then solving the production subproblem, given the quantity shipped to the retailers in each time instant. This algorithm can be described as follows.

Heuristic VMI-DP

1. Solve the subproblem $D(\hat{y})$, with $\hat{y}_1 = \sum_i (U_i - I_{i0} + r_{i1})$ and $y_t = \sum_i r_{it}$, $t > 1$. Let $\hat{x}$ be the vector of the obtained values of x_{it}, $i \in M$, $t \in T$.
2. Solve the subproblem $P(\hat{x})$.

The scheme of the two above algorithms can be applied to the second decomposition. The two new algorithms, referred to as heuristic VMI-PDP' and heuristic VMI-DP', are identical to the heuristic VMI-PDP and the heuristic VMI-DP, respectively, with the only exception that the subproblem $P'(\hat{x})$ is solved instead of $P(\hat{x})$ and $D'(\hat{y})$ is solved instead of $D(\hat{y})$.

We now show that, if subproblems $P(\hat{x})$ and $D(\hat{y})$ are optimally solved, then the cost obtained by the algorithm VMI-PDP is equal to the cost obtained by the algorithm VMI-DP. The same holds for the algorithm VMI-PDP' compared to the algorithm VMI-DP'.

Let $x^{D(\hat{y})}$ and $z^{D(\hat{y})}$ be the optimal solution and cost, respectively, of the subproblem $D(\hat{y})$, $\tilde{x}_{it} = U_i - I_{it-1} + r_{it}$ and $\tilde{y}_t = \sum_i (U_i - I_{it-1} + r_{it})$ be the value of $\hat{y}_t$ and $\hat{x}_{it}$ used in the step 1 of the algorithm VMI-PDP and of the algorithm VMI-DP, respectively, and $\bar{y}_t$ be a different given value of $\hat{y}_t$, $t \in T$. Note that $\sum_{k=1}^{t} \tilde{y}_k = \sum_{k=1}^{t} \sum_i \tilde{x}_{ik} = \sum_i (U_i - I_{i0} + \sum_{k=1}^{t} r_{ik})$. Then, the following lemmas hold.

Lemma 1. $z^{D(\tilde{y})} \le z^{D(\hat{y})}, \forall \hat{y}$.

Proof: If $\hat{y} = \tilde{y}$, then the constraints (18)–(20) are satisfied. This follows from the fact that, for each $t \in T$

$$B_t = \sum_{k=1}^{t} \tilde{y}_k - \sum_{k=1}^{t} \sum_{i \in M} x_{ik} = \sum_{i \in M} \left(U_i - I_{i0} + \sum_{k=1}^{t} r_{ik} \right) - \sum_{k=1}^{t} \sum_{i \in M} x_{ik} \ge 0$$

as the order-up-to level policy implies that $\sum_{k=1}^{t} \sum_i x_{ik} \le \sum_i (U_i - I_{i0} + \sum_{k=1}^{t} r_{ik})$. Therefore, $z^{D(\tilde{y})} \le z^{D(\hat{y})}\ \forall \hat{y}$. □

Lemma 2. *If $\sum_{k=1}^{t} \bar{y}_k \ge \sum_{k=1}^{t} \tilde{y}_k$, $t \in T$, then $x^{D(\bar{y})} = x^{D(\tilde{y})}$.*

Let $z^{\text{VMI-PDP}}$ be the cost obtained by applying the algorithm VMI-PDP and $z^{\text{VMI-DP}}$ be the cost obtained by applying the algorithm VMI-DP.

Theorem 1. $z^{\text{VMI-PDP}} = z^{\text{VMI-DP}}$.

Proof: Two cases can happen by optimally solving $P(\tilde{x})$: the optimal solution $y^{P(\tilde{x})}$ is equal to $\tilde{y}$ or is such that $\sum_{k=1}^{t} y_k^{P(\tilde{x})} \ge \sum_{k=1}^{t} \sum_i \tilde{x}_{ik}$, $t \in T$, thanks to the stock-out constraints at the production facility. In the former case, the optimal solution of $D(y^{P(\tilde{x})})$ is obviously $x^{D(\tilde{y})}$. In the latter case, since $\sum_{k=1}^{t} \sum_i \tilde{x}_{ik} = \sum_{k=1}^{t} \tilde{y}_k$, then $\sum_{k=1}^{t} y_k^{P(\tilde{x})} \ge \sum_{k=1}^{t} \tilde{y}_k$, $t \in T$. Therefore, $x^{D(\tilde{y})}$ is the optimal solution of $D(y^{P(\tilde{x})})$, thanks to Lemma 2. □

2.3. Solving the subproblems

Due to the results presented in the previous section, we will analyze only one hierarchical algorithm, namely the VMI-DP for the first decomposition and the VMI-DP' for the second decomposition. We start with the description of the algorithms for the subproblems of VMI-DP and then describe how these can be modified for the solution of the subproblems of VMI-DP'.

In Section 2.4 we will describe some improvement procedures which can be applied to improve the final solution.

The subproblem $P(\hat{x})$

The subproblem $P(\hat{x})$ is the classical uncapacitated dynamic lot size problem proposed by Wagner and Whitin (1958), which can be optimally solved in polynomial time by applying a procedure which is described, for instance, in Lee and Nahmias (1993). The procedure, which we describe here for the sake of completeness, works on an acyclic network $N(T', A, Q, P)$, in which each element of the set T' is a node that corresponds to a discrete time instant between 0 and $H+1$, each element a_{kt}, with $0 < k < t = 2, 3, \dots, H+1$, of the set A is an arc that exists if k and t are two possible consecutive production time instants and each element a_{0t} is an arc that exists if no shipments to the retailers are performed before time t, that is if $\sum_{j=1}^{t-1}\sum_i \hat{x}_{ij} = 0$. The intermediate nodes in each path on the network between 0 and $H+1$ represent a feasible set of production time instants. Each element q_{kt} of the set Q is a weight on the arc a_{kt} that represents the quantity to produce at time k in order to satisfy the stock-out constraints at the production facility in the time interval between k and $t-1$, that is $q_{kt} = \sum_{j=k}^{t-1}\sum_{i\in M} \hat{x}_{ij}$, for $0 < k < t = 2, 3, \ldots, H+1$, and $q_{0t} = 0$ for each arc a_{0t}, as 0 is not a production time instant. Finally, each element p_{kt} of the set P is a weight on the arc a_{kt} that represents the increase in the total cost to produce the quantity q_{kt} at time k, that is $p_{kt} = K + h_0(H+1-k)q_{kt} + pq_{kt}$ for $0 < k < t = 2, 3, \ldots, H+1$, and $p_{0t} = 0$ for all t. These weights are used to determine the shortest path between 0 and $H+1$ on the network by applying an algorithm for acyclic networks (see for instance Hu, 1982). The intermediate nodes on the shortest path are the optimal production time instants. If z^S denotes the cost of the shortest path, the optimal production cost z^{H^P} is given by the sum of z^S and of the constant part of the objective function.

The procedure can be formally described as follows.

Procedure H^P

(1) Set z^{H^P} equal to the constant part of the objective function, that is $z^{H^P} := -\sum_i \sum_t \sum_{j=1}^{t} h_0 \hat{x}_{it}$.
(2) Build the acyclic network $N(T', A, Q, P)$.
(3) Determine the shortest path between 0 and $H+1$ and compute the corresponding cost z^S on the basis of the weights in P. Set $z^{H^P} := z^{H^P} + z^S$.
(4) For each time instant $k \in T$, set the optimal production quantity $y_k^* := q_{kt}$ if the corresponding arc a_{kt} belongs to the shortest path, and $y_k^* := 0$ otherwise, and compute $B_k = B_{k-1} + y_k^* - \sum_{i\in M} \hat{x}_{ik}$.

The subproblem $D(\hat{y})$

Since the subproblem $D(\hat{y})$ is NP-hard, we propose a heuristic algorithm to solve it. A feasible solution of the problem is built by an iterative procedure that inserts a retailer at each iteration. When retailer i is considered, a set D_i of delivery time instants is determined by applying the procedure *Assign*. Then, for each of the selected delivery time instants $t \in D_i$, the retailer i is inserted in one of the routes determined for the vehicles traveling at time t by applying the procedure *Insert*. The algorithm can be formally described as follows.

Procedure H^D

(1) Sort the set of retailers M in the non-decreasing order of α_i, where α_i is the maximum integer number such that $\sum_{t=1}^{\alpha_i} r_{it} \leq I_{i0}$ and represents the number of time instants,

starting from time 1, for which the retailer i will not have stock-out even if not served. Set the value of the objective function of the subproblem $D(\hat{y})$ equal to its constant part, that is $z^{H^D} := \sum_i \sum_t h_i(I_{i0} - \sum_{j=1}^{t} r_{ij})$ and compute $B_t = B_{t-1} + \hat{y}_t$, $t \in T$ because no retailer is visited.

(2) For $s = 1, 2, \ldots, n$

- Determine for the retailer s a set D_s of delivery time instants by using the procedure *Assign*.
- For each time instant $t \in D_s$ insert the retailer s in one of the routes R_{tv}, $v \in V$, by using the procedure *Insert*.

Let us now describe the procedures applied during the algorithm. The procedure *Assign* determines a feasible set D_s of delivery time instants for the retailer s. It works on an acyclic network $G_s(T'_s, A_s, Q_s, P_s)$ in which each element of the set T'_s is a node that corresponds to a discrete time instant between 0 and $H+1$ and each element a^s_{kt} of the set A_s is an arc that exists if no stock-out occurs in s whenever s is not visited between k and t; therefore, the intermediate nodes of each path on the network between 0 and $H+1$ represents a set of delivery time instants for s that satisfy the stock-out constraints (7). Each element q^s_{kt} of the set Q_s is a weight on the arc a^s_{kt} that represents the quantity to deliver to s at time t and each element p^s_{kt} of the set P_s is a weight on the arc a^s_{kt} used in order to determine the shortest path between 0 and $H+1$ on the network, i.e. a "good" set of delivery time instants for s. Let us describe in more detail the sets A_s, Q_s and P_s. The elements of A_s are the arcs that satisfy the stock-out constraints (7) at the retailer s; in particular, the arc a^s_{0t}, $1 \le t \le H+1$, exists if $\sum_{j=1}^{t} r_{sj} \le I_{s0}$ and the arc a^s_{kt}, $1 \le k < t \le H+1$, exists if $\sum_{j=k+1}^{t} r_{sj} \le U_s$. Note that if the arc $a^s_{0,H+1}$ exists, then a feasible policy is not to visit the retailer during the time horizon. The set Q_s is a set of weights in which each element q^s_{kt}, associated to the arc a^s_{kt}, represents the quantity x_{st} to ship to s at time t determined on the basis of (5). Given that an order-up-to level policy is adopted, then the quantity q^s_{kt} is such that the maximum level of the inventory U_s is reached in s, that is $q^s_{kt} = \sum_{j=k+1}^{t} r_{sj}$ for each arc a^s_{kt} with $1 \le k < t \le H$, $q^s_{0t} = U_s - I_{s0} + \sum_{j=1}^{t} r_{sj}$ for each arc a^s_{0t} with $1 \le t \le H$ and $q^s_{k,H+1} = 0$, $0 \le k \le H$, given that a shipment cannot be performed in $H+1$. Finally, the set P_s is a set of weights in which each element p^s_{kt}, associated to the arc a^s_{kt}, represents the estimate of the variation in the total cost obtained by including in the current solution a visit of the retailer s at time t, given that the previous visit has been at time k. Given the partial solution, the weight p^s_{kt} on each arc a^s_{kt} is computed as the sum of two components. The first one $\tilde{c}^s_t$ is the estimate of the variation in the transportation cost obtained if the retailer s is served at time t and is computed as described in the procedure *Insert*. The second component of the weight p^s_{kt} is an estimate $\tilde{I}^s_{kt}$ of the variation in the inventory cost, which we set equal to $h_s \sum_{j=t}^{H} q^s_{kj} = h_s(H+1-t)q^s_{kt}$, as the constant part of the inventory cost at the retailer s is taken into account in step 1) of the procedure H^D. In conclusion, the weight p^s_{kt} associated to the arc a^s_{kt} is:

$$p^s_{kt} = \tilde{c}^s_t + \tilde{I}^s_{kt}. \tag{23}$$

Obviously, $p^s_{k,H+1} = 0$, as no shipments are performed at time $H+1$, and $p^s_{kt} = \infty$ if at time t the stock-out constraint at the production facility is violated. Once the weight p^s_{kt} is computed for each arc $a^s_{kt} \in A_s$, the procedure determines the shortest path between 0 and $H+1$, by using an algorithm for acyclic networks (see for instance Hu, 1982), in order to obtain a set of delivery time instants for s that minimize the incremental cost to adding the retailer s to the solution. Finally, the procedure includes in the set D_s of the selected delivery time instants for s the intermediate nodes that belong to the shortest path. The procedure *Assign* can be formally described as follows.

Procedure Assign

(1) Build the acyclic network $G_s(T'_s, A_s, Q_s, P_s)$.
(2) Determine the shortest path between 0 and $H+1$ on the basis of the weights in P_s.
(3) Include in the set D_s the intermediate nodes that belong to the shortest path.

The procedure *Insert* allows us to insert a given retailer s in one of routes R_{tv}, $v \in V$, traveled at time t on the basis of the cheapest insertion cost. It can be formally described as follows. Let $\hat{C}_{tv}$ be the current residual transportation capacity of vehicle v at time t, $su(i,t,v)$ be the successor of node $i \in R_{tv}$ in the route R_{tv} and Δ_{tv} be the variation in the transportation cost obtained by inserting s in the route R_{tv}.

Procedure Insert

(1) For each vehicle $v \in V$:

If $R_{tv} = \emptyset$, then $\Delta_{tv} = 2c_{0s}$ if v is already used in a different time instant, $\Delta_{tv} = f + 2c_{0s}$ otherwise.
Else, if $q^s_{kt} \le \hat{C}_{tv}$, then

$$\Delta_{tv} = \min_{i \in R_{tv}} \left\{ c_{i,s} + c_{s,su(i,t,v)} - c_{i,su(i,t,v)} \right\},$$

otherwise $\Delta_{tv} = \infty$.

(2) Select the vehicle v^* such that $v^* = \arg\min_{v\in V}\{\Delta_{tv}\}$.
(3) If $R_{tv^*} = \emptyset$, then $R_{tv^*} := \{0, s, 0\}$.
Else
- Determine $i^* = \arg\min_{i \in R_{tv^*}}\{c_{i,s} + c_{s,su(i,t,v^*)} - c_{i,su(i,t,v^*)}\}$.
- Remove from R_{tv^*} the arc $(i^*, su(i^*, t, v^*))$.
- Add to R_{tv^*} the arcs (i^*, s) and $(s, su(i^*, t, v^*))$.

The insertion of the retailer s in the route R_{tv^*} implies an increase in the total quantity loaded on the vehicle v^* equal to q^s_{kt} (i.e. $\hat{C}_{tv^*} := \hat{C}_{tv^*} - q^s_{kt}$), a variation in the inventory cost at the retailers equal to $h_s(H+1-t)q^s_{kt}$ and a variation in the transportation cost equal to Δ_{tv^*}. Therefore, $z^{H^D} := z^{H^D} + h_s(H+1-t)q^s_{kt} + \Delta_{tv^*}$. Finally, the insertion implies a variation in the level of the inventory at the production facility equal to $-q^s_{kt}$ for each time instant $j = t, t+1, \ldots, H$, that is $B_j := B_j - q^s_{kj}$ for $j = t, t+1, \ldots, H$.

The subproblems $P'(\hat{x})$ and $D'(\hat{y})$

The subproblem $P'(\hat{x})$ is identical to the subproblem $P(\hat{x})$, with the only exception of the objective function which does not contain the variable production cost and the part of the inventory cost at the production facility which depends on the variables x's only. Therefore, the optimal solution of the subproblem can be obtained by applying the procedure H^P, with $z^{H^P} := 0$ in the step 1) of the procedure, and $p_{kt} = K + h_0(H + 1 - k)q_{kt}$ in the acyclic network $N(T', A, Q, P)$.

The subproblem $D'(\hat{y})$ is identical to the subproblem $D(\hat{y})$, with the only exception of the objective function which contains the variable production cost and the part of the inventory cost at the production facility which depends on the variables x's only. Therefore, the subproblem can be solved by applying the procedure H^D with

$$p^s_{kt} = \tilde{c}^s_t + \tilde{v}^s_{kt} + \tilde{I}^s_{kt} \tag{24}$$

instead of (23) in the acyclic network $G_s(T'_s, A_s, Q_s, P_s)$. The estimate $\tilde{c}^s_t$ of the variation in the transportation cost is computed as in (23); the estimate $\tilde{v}^s_{kt}$ of the variation in the variable production cost is equal to pq^s_{kt} and the estimate $\tilde{I}^s_{kt}$ of the variation in the inventory cost is equal to the sum of the variation in the inventory cost at the retailer $h_s(H + 1 - t)q^s_{kt}$ and the variation in the part of the inventory cost at the production facility which depends on the shipping quantities, that is $-h_0(H + 1 - t)q^s_{kt}$.

2.4. Improving the obtained solution

The solution obtained by hierarchically solving the subproblems can be improved by applying the following two procedures, referred to as *Improve* and *Global Improve*. In the procedure *Improve*, the current solution is improved iteratively. At each iteration, two retailers are temporarily removed from the current solution by applying the procedure *Remove* described in the following. Then, the retailers are inserted in the current solution by applying the procedure *Assign*. Finally, the subproblem $P(\hat{x})$ is solved to determine the optimal quantity to produce in each time instant, given the quantity $\hat{x}_{it}$ to deliver to each retailer i in each time instant t. If this reduces the total cost, then the solution is modified accordingly. This iteration is repeated as long as an improvement in the total cost is reached. The procedure can be formally described as follows. Let TC be the total cost of the current solution.

Procedure Improve

(a) For $s = 1, 2, \ldots, n$
    For $i = n, n-1, \ldots, 1$ and $i \neq s$

(a1) Remove the retailer s from the routes R_{tv}, $v \in V$, by using the procedure *Remove* applied to all $t \in D_s$. Do the same for the retailer i.

(a2) Determine for i a new set $\tilde{D}_i$ of delivery time instants by using the procedure *Assign* and insert the retailer in one of the routes R_{tv}, $v \in V$, by using the procedure *Insert* applied to all $t \in \tilde{D}_i$. Do the same for the retailer s. Let $\hat{x}$ be the vector of the obtained values x_{it}, $i \in M$, $t \in T$.

(a3) Determine for each time instant $t \in T$ the new quantity y_t to produce by solving the subproblem $P(\hat{x})$. Let $\hat{y}$ be the vector of the obtained values y_t, $t \in T$.

(a4) Let $\bar{TC}$ be the cost of the obtained solution. If $\bar{TC} < TC$, then adopt the new solution.

(b) If a new solution has been adopted for at least one retailer, then go to (a). Otherwise, stop.

The procedure *Global Improve* tries to furtherly reduce the total cost by coordinating production and distribution. First, at time 0, the level of inventory at the production facility is set to the total quantity produced over the time horizon reduced by a fixed quantity $r^* = \min_{i \in M, t \in T} r_{it}$, i.e. $B_0 = \sum_{t \in T} \hat{y}_t - r^*$, and for each time instant $t \in T$ the level of inventory at the production facility is set equal to the level at time $t - 1$, minus the total quantity shipped to the retailers at time t, that is $B_t = B_{t-1} - \sum_{i \in M} \hat{x}_{it}$. Finally, the procedure *Improve* is applied, with the step a4) modified as follows:

(a4) Let $\bar{TC}$ be the cost of the obtained solution. If $\bar{TC} < TC$, then adopt the new solution. Compute $B_0 = \sum_{t \in T} \hat{y}_t - r^*$ and $B_t = B_{t-1} - \sum_{i \in M} \hat{x}_{it}$, for $t \in T$.

Let us now describe the procedure *Remove* used in the procedures *Improve* and *Global Improve* to remove the retailer s from a given route R_{tv}, $t \in D_s$. Two different situations can happen, depending on the fact that s is the only retailer visited in the route or not before removing it. Let $pr(s, t, v)$ and $su(s, t, v)$ be the predecessor and the successor of the retailer s in the route R_{tv}.

Procedure Remove

If $R_{tv} = \{0, s, 0\}$, then remove the arcs $(0, s)$ and $(s, 0)$.
Else
- Remove from R_{tv} the arcs $(pr(s, t, v), s)$ and $(s, su(s, t, v))$.
- Add to R_{tv} the arc $(pr(s, t, v), su(s, t, v))$.

The decrease in the total quantity of the item loaded on the vehicle is q^s_{kt}. The variation in the inventory cost at the retailers is $-h_s(H + 1 - t)q^s_{kt}$. If only the retailer s was in the route before removing it, the variation in the transportation cost is $\Delta_{tv} = -2c_{0s}$ if the vehicle is also used in a different time instant, while it is $\Delta_{tv} = -f - 2c_{0s}$ otherwise. If $R_{tv} \neq \emptyset$ after removing s, then the variation in the transportation cost is $\Delta_{tv} = c_{pr(s,t,v),su(s,t,v)} -$

$c_{pr(s,t,v),s} - c_{s,su(s,t,v)}$. Therefore, $z^{H^D} := z^{H^D} - h_s(H + 1 - t)q^s_{kt} + \Delta_{tv}$. Removing the retailer s implies a variation in the level of the inventory at the production facility equal to q^s_{kt} for each time instant $j = t, t+1, \ldots, H$, that is $B_j := B_j + q^s_{kt}$ for $j = t, t+1, \ldots, H$.

3. Relaxing the order-up-to level constraints: The fill-fill-dump policy

The problem formulated in Section 1 and solved in the previous section is based on the strict application of the order-up-to level policy. In other words, whenever a retailer is served, the order-up-to level quantity is always shipped to the retailer. This can imply both higher inventory cost at the retailers and lower saturation of the transportation capacity, and therefore higher transportation cost, with respect to different replenishment policies in which the order-up-to level constraints are at least partially relaxed. An example is given by the so called fill-fill-dump policy, in which the order-up-to level quantity is shipped to all but the last retailer on each delivery route, while the minimum between the order-up-to level quantity and the residual transportation capacity is shipped to the last retailer.

We have implemented two solution algorithms for the case the fill-fill-dump policy is applied, referred to as VMI-DP-F and VMI-DP'-F. They are identical to the order-up-to level algorithms VMI-DP and VMI-DP', respectively, with the exception of the procedure *Assign*, which is replaced by the procedure *Assign*^{F} described in the following, and of the procedures *Improve* and *Global Improve*, which are not applied at all, as the solution obtained by these procedures can be different from a fill-fill-dump solution. The procedure *Assign* cannot be applied here as the quantity delivered to the retailer s at each time t is not independent of the previous delivery time instants, but it depends on the residual transportation capacity of the vehicles used to serve the retailer s. Therefore, the level of the inventory at the production facility and at the retailer s at each time instant t depends on the previous delivery time instants and on the vehicles used to serve the retailer s up to time t.

The procedure *Assign*^{F} heuristically determines a set D_s of delivery time instants for the retailer s, the vehicles that perform the visits and the delivery quantities. A heuristic approach is needed as an exact algorithm is impractical for instances with long time horizon. The procedure works on an acyclic network G^F_s in which each intermediate node (t, v_t) corresponds to the visit to retailer s at time t by using the vehicle v_t. This vehicle can be one of the vehicles currently used at time t ($v_t = 1, \ldots, \hat{v}_t$) or a new vehicle ($v_t = \hat{v}_t + 1$). The nodes (0, 0) and $(H + 1, 0)$ are the first and the last node of the network, respectively, and do not correspond to any delivery time instant. When a fill-fill-dump policy is applied, the quantity shipped to the retailer s at time t by using the vehicle v_t is the minimum between the order-up-to level quantity and the residual transportation capacity C_{tv_t} of vehicle v_t. If we denote by k and t two consecutive delivery time instants and by I_{skv_k} the level of the inventory at the retailer s at time k when s is served at time k by the vehicle v_k, then the quantity delivered at time t by the vehicle v_t is $q^s_{kv_k,tv_t} = \min\{U_s - I_{skv_k} + \sum_{j=k+1}^{t} r_{sj}, C_{tv_t}\}$, where $I_{s00} = I_{s0}$. The corresponding estimate $p^s_{kv_k,tv_t}$ of the variation in the total cost is computed as indicated in Section 2.3, by simply replacing q^s_{kt} with $q^s_{kv_k,tv_t}$ in the procedure *Assign* and by setting the estimate of the variation in the transportation cost in the procedure

Insert equal to Δ_{tv} instead of Δ_{tv^*}. Obviously, $p^s_{kv_k,tv_t} = +\infty$ if at least one of the following three conditions is verified. First, the residual transportation capacity C_{tv_t} of the vehicle v_t used at time t is lower than the quantity absorbed by the retailer s at time t (i.e. $C_{tv_t} < r_{st}$). Second, the stock-out constraint at the production facility is not satisfied (i.e. the level B_{stv_t} of the inventory at the production facility when the retailer s is served at time t by the vehicle v_t is negative). Third, the stock-out constraint at the retailer s is not satisfied (i.e. $I_{stv_t} < 0$).

The procedure can be described as follows. Let u_{tv_t} be the cost of the best known path between $(0, 0)$ and (t, v_t). At the beginning, $u_{tv_t} = p^s_{00,tv_t}$ for each intermediate node (t, v_t) and for the node $(H + 1, 0)$. The procedure iteratively determines the "best" path between the node $(0, 0)$ and each node (t, v_t) and updates u_{lv_l}, for $l > t$, to the minimum between its current value and the sum of u_{tv_t} and the cost $p^s_{tv_t,lv_l}$ on the arc between the nodes (t, v_t) and (l, v_l). Finally, whenever u_{lv_l} is modified, it updates $q^s_{lv_l}$, I_{slv_l} and B_{slv_l}.

Procedure Assign^F

(1) Build the set of nodes of the network G^F_s, that is the nodes $(0, 0)$, (t, v_t) for $t = 1, \ldots, H$ and $v_t = 1, \ldots, \hat{v}_t + 1$, and $(H + 1, 0)$.
(2) Compute the total cost of never visiting the retailer s during the time horizon, i.e. the cost of the path which includes the nodes $(0, 0)$ and $(H + 1, 0)$ only. Update $u_{H+1,0}$. For each intermediate node (t, v_t), with $t = 1, \ldots, H$ and $v_t = 1, \ldots, \hat{v}_t + 1$, compute the total cost of the path which includes the nodes $(0, 0)$ and (t, v_t) only. Update u_{tv_t}, $q^s_{tv_t}$, I_{stv_t} and B_{stv_t}.
(3) For each node (t, v_t), with $t = 1, \ldots, H$ and $v_t = 1, \ldots, \hat{v}_t + 1$: for each node (l, v_l), with $l > t$, compute $p^s_{tv_t,lv_l}$; if $u_{lv_l} > u_{tv_t} + p^s_{tv_t,lv_l}$, then, update u_{lv_l}, $q^s_{lv_l}$, I_{slv_l} and B_{slv_l}, considering (t, v_t) as predecessor of (l, v_l).
(4) For each node (t, v_t), with $t = 1, \ldots, H$ and $v_t = 1, \ldots, \hat{v}_t + 1$, compute $p^s_{tv_t,(H+1)0}$; if $u_{H+1,0} > u_{tv_t} + p^s_{tv_t,(H+1)0}$, then update $u_{H+1,0}$.
(5) Include in the set D_s the intermediate nodes of the path between $(0, 0)$ and $(H + 1, 0)$ having cost $u_{H+1,0}$.

4. Computational results

The algorithms described in Sections 2.2 and 3, for the order-up-to level policy and for the fill-fill-dump policy, respectively, have been implemented in Fortran and compared on the basis of randomly generated problem instances. The computational results allow us to evaluate the impact of the different decompositions and of the different heuristics and to compare the application of both order-up-to level and fill-fill-dump policies with respect to a conventional Retailer-Managed Inventory policy, referred to as RMI. In the RMI policy, all the retailers that will have stock-out at time $t + 1$ are visited at time t. The routes are computed by applying for each time t the procedure *Insert*. The optimal production quantities are determined by solving the subproblem $P(\hat{x})$. Finally, the transportation cost is reduced, if possible, by applying a variant of the procedure *Improve*, in which the delivery time instants are not modified during the algorithm.

Four classes of instances have been generated, corresponding to different practical situations. The first class is composed of 24 instances (1–24), generated as follows:

Quantity r_{it} absorbed by retailer i at time t: Constant over time, i.e. $r_{it} = r_i$, $t \in T$, and randomly generated as an integer number in the interval [5, 25];
Maximum level U_i of the inventory at retailer i: $r_i g_i$, where g_i is randomly selected from the set S (as defined below) and represents the number of time units needed in order to consume the quantity U_i;
Starting level I_{i0} of the inventory at the retailer i: $U_i - r_i$;
Inventory cost at retailer $i \in M$, h_i: Randomly generated in the intervals [1, 5] and [6, 10];
Inventory cost at the production facility h_0: 3 and 8;
Variable production cost p: $10h_0$;
Fixed set-up cost K: $100p$;
Transportation cost c_{ij}: $\lfloor\sqrt{(X_i - X_j)^2 + (Y_i - Y_j)^2}\rfloor$, where the points (X_i, Y_i) and (X_j, Y_j), with $i \in M'$ and $j \in M'$, are obtained by randomly generating each coordinate as an integer number in the interval [0, 500] and in the interval [0, 1000];
Transportation capacity C: $\bar{U}$, $3/2\bar{U}$ and $2\bar{U}$, where $\bar{U} = \max_{i \in M} U_i$;
Fixed transportation cost $f = (n + 1) \max_{i,j \in M'} c_{ij}$.

The remaining three classes of instances have been generated as in the first class, with the exception of one type of cost only. In particular, the second class (instances 25–48) has been generated by setting the fixed transportation cost f to zero. This class allows us to evaluate the behaviour of the algorithms when the transportation is outsourced or the production facility has a given fleet of vehicles for distribution. In the third class (instances 49–72) the variable production cost p is set to h_0. This class allows us to evaluate the impact of different production costs. Finally, the fourth class (instances 73–96) is obtained by setting the inventory cost at the retailers h_i to zero, $i \in M$ and allows us to evaluate if the cases in which the production facility does not have to pay the inventory costs at the retailers have substantially different solutions.

In all cases, random selections have been performed in accordance to a uniform distribution. The computations have been carried out on an Intel Pentium III with 933 Mhz and 256 MB RAM. The generated instances and the computational results are available at the following URL: www-c.eco.unibs.it/~bertazzi/pd.zip.

A detailed computational analysis of the policies discussed in this paper will be presented for a number of retailers $n = 50$ and a time horizon $H = 30$. For these instances we choose $S = \{2, 3, 5, 6\}$. Then, more synthetic results will be presented for $n = 50, 75, 100, 125, 150$ and $H = 6$, with $S = \{2, 3, 6\}$.

We first compare the order-up-to level policy with the RMI policy. Average results on all the instances are shown in Tables 1–4. Table 1 is organized as follows. The first column gives the description of the type of result (average percent error, maximum percent error, number of best solutions and computational time) shown on the corresponding row. Three rows are dedicated to each type of result. The first row gives the results obtained by applying

Table 1. Percent errors and statistics.

	RMI	VMI-DP	VMI-DP'
Average error without *Improve* and *Global Improve*	54.98	4.70	5.99
Average error without *Global Improve*	51.31	2.03	3.55
Average error	51.31	0.02	1.80
Maximum error without *Improve* and *Global Improve*	180.34	10.90	24.47
Maximum error without *Global Improve*	180.34	10.90	22.63
Maximum error	180.34	0.38	21.46
Number of best solutions without *Improve* and *Global Improve*	0	61	35
Number of best solutions without *Global Improve*	0	78	18
Number of best solutions	0	86	10
Time without *Improve* and *Global Improve*	0.2	0.1	0.2
Time without *Global Improve*	117.9	228.1	288.8
Time	117.9	228.1	288.8

the algorithms RMI, VMI-DP and VMI-DP', respectively, without the procedures *Improve* and *Global Improve*, the second the results obtained by applying the algorithms without the procedure *Global Improve* and, finally, the third row the results obtained by applying the entire algorithms. Columns 2 gives the results obtained by the algorithm RMI (note that, for each type of result, the second and the third rows are identical, as in the RMI algorithm the procedure *Global Improve* is not applied). Finally, columns 3–4 give the results obtained by the VMI-DP and VMI-DP' algorithms, respectively. The errors are always computed with respect to the best obtained solution.

The results show that the best solutions are obtained when the first decomposition is applied, i.e. VMI-DP is better than VMI-DP'. The algorithm VMI-DP gives the best average and maximum percent relative error, finds the best solution in the 89.58% of the instances and has the lowest computational time. Moreover, the RMI policy is significantly outperformed by the algorithms VMI-DP and VMI-DP' in terms of quality of the solution. Finally, the procedures *Improve* and *Global Improve* significantly reduce the average and maximum error generated by the algorithms. Note that the major part of the computational time is spent by the procedure *Improve*.

Table 2 compares the composition of the total cost in the solutions generated by the algorithms RMI, VMI-DP and VMI-DP'.

The results show that, as expected, the poor performance of the algorithm RMI with respect to the algoritms VMI-DP and VMI-DP' is mainly due to the lack of coordination in the RMI policy, which generates a large transportation cost. Note that in the VMI-DP and VMP-DP' algorithms each cost component is reduced when the procedure *Improve* is applied, while in the procedure *Global Improve* all the cost components are reduced, with the only exception of the inventory cost at the production facility, which slightly increases.

Table 2. Composition of the total cost.

	RMI	VMI-DP	VMI-DP'
Inv. cost prod. facility without *Improve* and *Global Improve*	18364.50	35618.34	34480.48
Inv. cost prod. facility without *Global Improve*	18364.50	31137.92	29387.47
Inv. cost prod. facility	18364.50	31513.80	29592.91
Production cost without *Improve* and *Global Improve*	1029779.00	954302.30	943118.30
Production cost without *Global Improve*	1029779.00	941868.30	935855.10
Production cost	1029779.00	936045.20	934160.70
Inv. cost retailers without *Improve* and *Global Improve*	227042.00	231946.50	244248.70
Inv. cost retailers without *Global Improve*	227042.00	229756.80	241882.70
Inv. cost retailers	227042.00	227844.60	239767.60
Transportation cost without *Improve* and *Global Improve*	883722.10	322382.30	335502.40
Transportation cost without *Global Improve*	826432.20	298537.20	311280.40
Transportation cost	826432.20	286292.60	299511.30

Table 3. Production vs distribution.

	RMI	VMI-DP	VMI-DP'
Total production cost	1048143.50	967559.00	963753.61
Total distribution cost	1053474.20	514137.20	539278.90

Table 3 compares the total production cost (inventory at the production facility plus fixed and variable production costs) with the total distribution cost (inventory at the retailers plus fixed transportation and routing costs) in the solutions generated by the algorithms RMI, VMI-DP and VMI-DP'.

The results show that in the solution generated by the RMI algorithm the total production cost is about 50% of the total cost, while in the VMI-DP and VMI-DP' policies it is about 65% of the total cost. Moreover, the worse performance of the VMI-DP' with respect to the VMI-DP is due to an increase in the total distribution cost not sufficiently compensated by the reduction of the total production cost.

Table 4 compares the solution generated by the algorithms RMI, VMI-DP and VMI-DP' on the basis of the average production and shipping quantity (denoted by Quantity), the average number of visits (Visits), the average number of production time instants (Production times) and, finally, the average number of vehicles (Vehicles).

Let us first compare VMI-DP with respect to VMI-DP'. The results show that the number of visits tends to be significantly greater when VMI-DP' is applied. This is due to the inclusion in the objective function of the subproblem $D'(\hat{y})$ of the part of the inventory cost of the production facility which depends on the variables x's only. This part tends to make more appealing solutions in which the item is shipped as soon as possible. Therefore, small

Table 4. Average quantities.

	RMI	VMI-DP	VMI-DP'
Quantity without *Improve* and *Global Improve*	22830.00	20917.02	20728.83
Quantity without *Global Improve*	22830.00	20702.33	20615.73
Quantity	22830.00	20536.83	20510.07
Visits without *Improve* and *Global Improve*	426.00	474.94	557.80
Visits without *Global Improve*	426.00	477.68	599.57
Visits	426.00	479.65	617.33
Production times without *Improve* and *Global Improve*	15.25	18.13	18.20
Production times without *Global Improve*	15.25	18.02	17.89
Production times	15.25	17.63	17.71
Vehicles without *Improve* and *Global Improve*	19.33	6.27	5.64
Vehicles without *Global Improve*	18.63	6.19	5.94
Vehicles	18.63	6.16	5.93

quantities tend to be more frequently shipped to the retailers, as this reduces the total inventory cost of the subproblem $D'(\hat{y})$. The number of vehicles reduces, as the transportation capacity is better managed when small quantities are shipped. However, since the number of visits increases, the inventory cost at the retailers and the routing cost increase (see Table 2). Note that the increase in the routing cost is not compensated by the reduction of the fixed transportation cost. Therefore, the total distribution cost increases in the VMI-DP' with respect to the VMI-DP (as shown in Table 3). Moreover, since the number of visits increases and the number of production time instants does not significantly change, the inventory cost at the production facility reduces (see Table 2). Finally, the introduction in the objective function of the subproblem $D'(\hat{y})$ of the variable production cost tends to reduce the production and distribution quantity. Therefore, the total production cost reduces in the VMI-DP' with respect to the VMI-DP (as shown in Table 3). Let us now compare the RMI policy with respect to the VMI policies. The most relevant difference between the RMI and the VMI-DP and VMI-DP' is in the number of vehicles which is much greater in the former, due to the lack of coordination of the retailer requests. The number of visits is instead lower in the RMI, as well as the inventory cost (see Table 2). However, the reduction of the inventory costs is far from compensating the increase in the transportation costs. Moreover, in the VMI-DP and VMI-DP' solutions the production and shipping quantity as well as the number of production time instants reduce when the procedure *Improve* and *Global Improve* are applied, while the number of visits increases.

With Tables 5–8 we compare the behaviour of the solutions in the four classes of instances. In particular, Table 5 shows, for each class of instances, the average percent error, Table 6 the composition of the total cost, Table 7 compares the total production cost with the total distribution cost and Table 8 shows the production and shipping quantity, the number of visits, the number of production time instants and the number of vehicles obtained.

Table 5. Classes of instances: Average percent error.

Instances	RMI	VMI-DP	VMI-DP'
1–24	42.69	0.06	0.85
25–48 ($f = 0$)	9.02	0.01	1.27
49–72 ($p = h_0$)	100.48	0.02	4.08
73–96 ($h_i = 0$)	53.04	0.00	1.01

Table 6. Classes of instances: Composition of the total cost.

Instances	Cost	RMI	VMI-DP	VMI-DP'
1–24	Inventory cost production facility	24486.00	51759.58	50290.21
	Production cost	1327150.00	1208960.00	1204601.00
	Inventory cost retailers	302722.50	308724.70	322449.50
	Transportation cost	1024906.00	341743.70	351922.50
25–48 ($f = 0$)	Inventory cost production facility	24486.00	22378.88	21956.33
	Production cost	1327150.00	1200741.00	1201385.00
	Inventory cost retailers	302722.50	294895.30	312876.30
	Transportation cost	138511.50	124484.10	129519.10
49–72 ($p = h_0$)	Inventory cost production facility	0.00	0.00	0.00
	Production cost	137665.00	128956.00	129638.80
	Inventory cost retailers	302722.50	307758.60	323744.60
	Transportation cost	1117404.00	339151.80	349097.50
73–96 ($h_i = 0$)	Inventory cost production facility	24486.00	51916.75	46125.08
	Production cost	1327150.00	1205523.00	1201018.00
	Inventory cost retailers	0.00	0.00	0.00
	Transportation cost	1024906.00	339790.70	367506.00

Table 7. Classes of instances: Production vs distribution.

Instances	Cost	RMI	VMI-DP	VMI-DP'
1–24	Total production cost	1351636.00	1260719.58	1254891.21
	Total distribution cost	1327628.50	650468.40	674372.00
25–48 ($f = 0$)	Total production cost	1351636.00	1223119.88	1223341.33
	Total distribution cost	441234.00	419379.40	442395.40
49–72 ($p = h_0$)	Total production cost	137665.00	128956.00	129638.80
	Total distribution cost	1420126.50	646910.40	672842.10
73–96 ($h_i = 0$)	Total production cost	1351636.00	1257439.75	1247143.08
	Total distribution cost	1024906.00	339790.70	367506.00

Table 8. Classes of instances: Average quantities.

	Instances	RMI	VMI-DP	VMI-DP'
Quantity	1–24	22830.00	20571.67	20501.29
	25–48 ($f = 0$)	22830.00	20396.58	20396.17
	49–72 ($p = h_0$)	22830.00	20648.58	20738.08
	73–96 ($h_i = 0$)	22830.00	20530.50	20404.75
Visits	1–24	426.00	473.25	562.58
	25–48 ($f = 0$)	426.00	442.75	539.21
	49–72 ($p = h_0$)	426.00	468.50	573.29
	73–96 ($h_i = 0$)	426.00	534.08	794.25
Production times	1–24	13.00	14.13	14.13
	25–48 ($f = 0$)	13.00	14.38	14.46
	49–72 ($p = h_0$)	22.00	28.00	28.00
	73–96 ($h_i = 0$)	13.00	14.00	14.25
Vehicles	1–24	18.33	4.33	4.42
	25–48 ($f = 0$)	18.50	11.63	10.29
	49–72 ($p = h_0$)	19.33	4.33	4.42
	73–96 ($h_i = 0$)	18.33	4.33	4.58

In the second class (instances 25–48) the fixed transportation cost f is set to zero. The results show, as expected, that in the VMI-DP and VMI-DP' solutions the number of used vehicles significantly increases. The results clearly show that the major part of the transportation cost is due to the fixed cost. Therefore, the RMI policy has in this class its best performance in terms of average error. In the third class (instances 49–72) the variable production cost p is set to h_0 instead of $10h_0$. Therefore, the fixed production cost $K = 100p$ is also significantly reduced. The results show that the number of production time instants is doubled and the inventory cost at the production facility is reduced to zero. Finally, in the fourth class (instances 73–96) the inventory cost at the retailers h_i is set to zero, $\forall i$. The results show that, as expected, the number of visits significantly increases, in particular in the VMI-DP' solution.

Let us now compare the results obtained by the order-up-to level policies with the results obtained by the fill-fill-dump policies. Table 9 shows the average percent error on all instances between the solution generated by each algorithm and the best solution. The algorithms VMI-DP and VMI-DP' are applied without the procedures *Improve* and *Global Improve*, while the best solution is computed among the solutions given by the entire VMI-PD and VMI-PD' algorithms and the VMI-PD-F and VMI-DP'-F algorithms. Table 10 shows the composition of the total cost, Table 11 compares the total production cost with the total distribution cost and Table 12 shows the production and shipping quantity, the number of visits, the number of production time instants and the number of vehicles.

Table 9. Order-up-to level vs fill-fill-dump: Percent errors and statistics.

	VMI-DP	VMI-DP'	VMI-DP-F	VMI-DP'-F
Average error	4.71	6.01	3.78	4.70
Maximum error	10.90	24.47	8.21	17.32
Number of best solutions	14	12	43	27
Time	0.1	0.2	0.6	0.0

Table 10. Order-up-to level vs fill-fill-dump: Composition of the total cost.

	VMI-DP	VMI-DP'	VMI-DP-F	VMI-DP'-F
Inventory cost production facility	35618.34	34480.48	35894.03	34796.77
Production cost	954302.30	943118.30	956709.30	947401.80
Inventory cost retailers	231946.50	244248.70	215789.30	226966.10
Transportation cost	322382.30	335502.40	325854.20	331699.50

Table 11. Order-up-to level vs fill-fill-dump: Production vs distribution.

	VMI-DP	VMI-DP'	VMI-DP-F	VMI-DP'-F
Total production cost	989920.64	977598.78	992603.33	982198.57
Total distribution cost	554328.80	579751.10	541643.50	558665.60

Table 12. Order-up-to level vs fill-fill-dump: Average quantities.

	VMI-DP	VMI-DP'	VMI-DP-F	VMI-DP'-F
Quantity	20917.02	20728.83	20962.13	20818.80
Visits	474.94	557.80	492.17	548.18
Production times	18.13	18.20	18.42	18.34
Vehicles	6.27	5.64	6.02	5.97

The results show that, as expected, the fill-fill-dump policy obtains better solutions with respect to the order-up-to level policies. It generates the best solution in about 73% of the instances with a reduction of the average error of about 1%, obtained by significantly reducing the total distribution cost (see Table 11). As expected, the reduction of the total distribution cost is mainly due to the reduction of the inventory cost at the retailers (see Table 10).

Finally, in Table 13 we present the results obtained for $n = 50, 75, 100, 125, 150$ and $H = 6$, with $S = \{2, 3, 6\}$. For each value of n, 96 instances have been generated according to the scheme described at the beginning of this section. The algorithms RMI and VMI-DP

Table 13. Average and maximum percent error with $H = 6$.

n	RMI	VMI-DP	VMI-DP-F
50	140.05 (248.79)	18.49 (43.94)	3.81 (12.54)
75	127.31 (221.37)	13.21 (36.17)	2.79 (8.60)
100	137.05 (233.64)	14.08 (29.47)	1.73 (8.54)
125	149.05 (245.70)	14.88 (31.31)	1.38 (8.05)
150	160.58 (256.94)	12.93 (31.42)	0.98 (9.11)

are applied without the improvement procedures and the errors are calculated with respect to the best solution found, where the best solution is computed among the solutions obtained by the entire RMI, the entire VMI-PD and the VMI-PD-F algorithms. The table shows the average percent errors and, in parentheses, the maximum errors.

The VMI-DP-F has always found the best solution and the computational time required by each of the algorithms has always been lower than one second. The reduction of the time horizon from 30 to 6 has caused an increase of the impact of the transportation cost on the total cost. This explains the larger errors shown in Table 13 when compared with the errors shown for the instances with $H = 30$.

Conclusions

In this paper we have considered a complex inventory routing problem, in which items are produced at (or distributed from) a facility to a set of retailers. The costs considered are fixed and variable production (or ordering) costs, transportation costs and inventory costs at the facility and at the retailers. The decisions to be taken are: When and how much to produce (or to order from an external supplier) at the facility and when, how much and to which retailers to ship. Different types of VMI policies can be implemented. We considered two of them: The first one, called order-up-to level policy, is such that the order-up-to level quantity is shipped to each retailer whenever it is served. The second one, called fill-fill-dump policy, is obtained by partially relaxing the order-up-to level constraints. The optimal RMI and VMI solutions are hard to obtain, since the problems to be solved include the vehicle routing problem which is well known as a very hard problem to optimally solve. It turns out that the VMI policies, although not optimally but heuristically found, can dramatically reduce the costs. The computational results show that the VMI policies significantly reduce the average error with respect to the RMI policy. While the inventory costs at the retailers are slightly lower in the RMI policy, the transportation costs are much greater and involve a much larger number of vehicles than in the VMI policies. Moreover, it turns out that the fill-fill-dump policies reduce the total cost, by reducing the total distribution cost. Our conclusion is that the coordination costs necessary to implement VMI policies are likely to be more than compensated by the cost reduction obtained by the opportunity to optimally manage an integrated system. Future research will be devoted to improve the fill-fill-dump policies and to implement more sophisticated VMI policies in which the order-up-to level constraints are furtherly relaxed.

Acknowledgments

The authors wish to thank two anonymous referees for their useful comments and suggestions.

References

Axsäter, S. (2001). "A Note on Stock Replenishment and Shipment Scheduling for Vendor-Managed Inventory Systems." *Management Science* 47, 1306–1310.

Axsäter, S. and K. Rosling. (1994). "Multi-Level Production-Inventory Control: Material Requirements Planning or Reorder Point Policies?" *European Journal of Operational Research* 75, 405–412.

Berman, O. and R.C. Larson. (2001). "Deliveries in an Inventory/Routing Problem Using Stochastic Dynamic Programming." *Transportation Science* 35, 192–213.

Bertazzi, L., G. Paletta, and M.G. Speranza. (2002). "Deterministic Order-up-to Level Policies in an Inventory Routing Problem." *Transportation Science* 36, 119–132.

Blumenfeld, D.E., L.D. Burns, J.D. Diltz, and C.F. Daganzo. (1985). "Analyzing Trade-offs between Transportation, Inventory and Production Costs on Freight Networks." *Transportation Research* 19B, 361–380.

Campbell, A.M., L. Clarke, A.J. Kleywegt, and M.W.P. Savelsbergh. (1998). "The Inventory Routing Problem." In T.G. Crainic and G. Laporte (eds.), *Fleet Management and Logistics*, Boston: Kluwer Academic Publishers, pp. 95–113.

Campbell, A.M. and M.W.P. Savelsbergh. (2004). "A Decomposition Approach for the Inventory Routing Problem." *Transportation Science* 38, 488–502.

Çetinkaya, S. and Lee, C.-Y. (2000). "Stock Replenishment and Shipment Scheduling for Vendor-Managed Inventory Systems." *Management Science* 46, 217–232.

Chandra, P. and M.L. Fisher. (1994). "Coordination of Production and Distribution Planning." *European Journal of Operational Research* 72, 503–517.

Cheung, K.L. and H.L. Lee. (2002). "The Inventory Benefit of Shipment Coordination and Stock Rebalancing in a Supply Chain." *Management Science* 48, 300–306.

Cohen, M.A. and H.L. Lee. (1988). "Strategic Analysis of Integrated Production-Distribution Systems: Models and Methods." *Operations Research* 35, 216–228.

Erengüç, Ş.S., N.C. Simpson, and A.J. Vakharia. (1999). "Integrated Production/Distribution Planning in Supply Chains: An Invited Review." *European Journal of Operational Research* 115, 219–236.

Fry, M.J., R. Kapuscinski, and T.L. Olsen. (2001). "Coordinating Production and Delivery under a (z, Z)-Type Vendor-Managed Inventory Contract." *Manufacturing & Service Operations Management* 3, 151–173.

Hall, N.G. and C.N. Potts. (2003). "Supply Chain Scheduling: Batching and Delivery." *Operations Research* 51, 566–584.

Hu, T.C. (1982). *Combinatorial Algorithms*, Reading, MA: Addison Wesley.

Lee, H.L. and S. Nahmias. (1993). "Single-Product, Single-Location Models." In *Handbooks in Operations Research & Management Science*, vol. 4, North-Holland: Elsevier Science Publishers, pp. 3–55.

Kleywegt, A.J., V.S. Nori, and M.W.P. Savelsbergh. (2002). "The Stochastic Inventory Routing Problem with Direct Deliveries." *Transportation Science* 36, 94–118.

Kleywegt, A.J., V.S. Nori, and M.W.P. Savelsbergh. (2004). "Dynamic Programming Approximation for a Stochastic Inventory Routing Problem." *Transportation Science* 38, 42–70.

Rabah, M.Y. and H.S. Mahmassani. (2002). "Impact of Electronic Commerce on Logistics Operations: A Focus on Vendor Managed Inventory (VMI) Strategies." Technical Report N. SWUTC/02/167227-1, Center for Transportation Research, University of Texas at Austin.

Sarmiento, A.M. and R. Nagi. (1999). "A Review of Integrated Analysis of Production-Distribution Systems." *IIE Transactions* 31, 1061–1074.

Thomas, D.J. and P.M. Griffin. (1996). "Coordinated Supply Chain Management." *European Journal of Operational Research* 94, 1–15.

Toth, P. and D. Vigo. (2002). *The Vehicle Routing Problem*, SIAM Monographs on Discrete Mathematics and Applications, Philadelphia.

Wagner, H.M. and T.M. Whitin. (1958). "Dynamic Version of the Economic Lot Size Model." *Management Science* 5, 89–96.

Webb, I.R. and R.C. Larson. (1995). "Period and Phase of Customer Replenishment: A New Approach to the Strategic Inventory/Routing Problem." *European Journal of Operational Research* 85, 132–148.