

Production, Manufacturing and Logistics

# Incorporating inventory and routing costs in strategic location models

Zuo-Jun Max Shen <sup>a,\*</sup>, Lian Qi <sup>b</sup>

<sup>a</sup> *Department of Industrial Engineering and Operations Research, University of California, 4141 Etcheverry Hall,  
Berkeley, CA 94720-1777, USA*

<sup>b</sup> *Department of Industrial and Systems Engineering, University of Florida, Gainesville, FL, USA*

Received 9 March 2005; accepted 25 March 2006  
Available online 22 May 2006

---

## Abstract

We consider a supply chain design problem where the decision maker needs to decide the number and locations of the distribution centers (DCs). Customers face random demand, and each DC maintains a certain amount of safety stock in order to achieve a certain service level for the customers it serves. The objective is to minimize the total cost that includes location costs and inventory costs at the DCs, and distribution costs in the supply chain. We show that this problem can be formulated as a nonlinear integer programming model, for which we propose a Lagrangian relaxation based solution algorithm. By exploring the structure of the problem, we find a low-order polynomial algorithm for the nonlinear integer programming problem that must be solved in solving the Lagrangian relaxation sub-problems. We present computational results for several instances of the problem with sizes ranging from 40 to 320 customers. Our results show the benefits of having an integrated supply chain design framework that includes location, inventory, and routing decisions in the same optimization model.

© 2006 Elsevier B.V. All rights reserved.

**Keywords:** Location models; Vehicle routing; Inventory; Integrated supply chain design models

---

## 1. Introduction

Fierce competition in today's global market forces companies to better design and manage their supply chain networks. There are roughly three different levels of decisions in a supply chain: the *strategic*, *tactical*, and *operational* levels. The decisions at different levels are typically treated separately in the literature. For example, most well-studied location models do not consider inventory costs, and shipment costs are estimated by direct shipping. Although one may argue that tactical inventory replenishment decisions and shipment schemes are not at the strategic level, and we should not consider them in the strategic planning phase,

---

\* Corresponding author. Tel.: +1 510 643 2392; fax: +1 510 642 1403.  
E-mail address: [shen@ieor.berkeley.edu](mailto:shen@ieor.berkeley.edu) (Z.-J. Max Shen).

however, failure to take the related inventory and shipment costs into consideration when deciding the locations of facilities can lead to sub-optimality, since strategic location decisions have a big impact on inventory and shipment costs (for details, see Section 5.3).

In this paper, we propose a supply chain design model, which considers the impacts of the strategic facility location decisions on the tactical inventory and shipment decisions. Specifically, we consider a three-tiered supply chain system consisting of one or more suppliers, distribution centers (DCs) and customers. We assume each customer has uncertain demand that follows a certain probability distribution. We assume the customers are uniformly scattered in a connected area. Several DCs will be opened and each DC is served directly by the supplier and distributes products to customers. Each customer will order at the beginning of the period, the DC combines the orders from different customers and order from the supplier. The number and locations of DCs are not given a priori. We assume that at every DC, there is a delivery truck with fixed capacity. The problem is to determine how many DCs to locate, where to locate them, which DC should serve which customers, how often to reorder at a DC and what level of safety stock to maintain, so as to minimize total location, shipment, and inventory costs, while ensuring a pre-specified level of service. A key problem is that the demand that is seen by each DC is a function of the demands at the customers assigned to that DC, which is a function of the assignment of customers to the DC. We assume the transportation costs and the inventory costs exhibit economies of scale under which the average unit cost decreases as the total volume of activity increases. This realistic assumption will result in several nonlinear terms in the formulation that we will present later.

The rest of this paper is organized as follows. Section 2 reviews some related models in the literature. Section 3 describes our integrated stochastic supply chain design model and Section 4 proposes a solution algorithm for the model. Computational results that highlight the effectiveness of the algorithm are reported in Section 5, where we also show the benefit of integration in supply chain design. Finally, in Section 6 we conclude the paper and discuss extensions and future research directions.

## 2. Related research

As we discussed in Section 1, there are three important decisions within a supply chain: facilities *location* decisions; *inventory* management decisions; and *distribution* decisions. It is clear that these three key elements of a supply chain are highly related. For example, an effective distribution scheme depends on the locations of facilities, and a good inventory management scheme depends on effective shipment plans. Being able to build a decision support system which integrates these elements of a supply chain is a major challenge and can provide a company with a tremendous competitive advantage in the market. In the literature, we have seen many papers that study the integration and coordination of any two of the above three important decisions: location-routing models, inventory-routing models, and location-inventory models.

For reviews on location-routing models, readers can refer to Balakrishnan et al. (1987) and Min et al. (1998). For inventory-routing models, please refer to Kleywegt et al. (2002a,b) and Adelman (2003).

There are many papers that study the location, inventory, and distribution coordination issues, but except the papers by Erlebacher and Meller (2000), Daskin et al. (2002) and Shen et al. (2003), most of the other papers either ignore the (nonlinear) inventory costs, or approximate the costs with linear functions. Erlebacher and Meller (2000) formulate a highly nonlinear integer location/inventory model. They attack the problem by using a continuous approximation as well as a number of construction and bounding heuristics. Computation times on a 600 node problem using an exchange heuristic averaged 117 hours on a Sun Ultra Sparcstation. Shen (2000), Shen et al. (2003), and Daskin et al. (2002) studied the joint location/inventory model in which location, shipment and nonlinear safety stock inventory costs are included in the same model. They developed an integrated approach to determine the number of DCs to establish, the location of the DCs, and the magnitude of inventory to maintain at each center. Shen et al. (2003) use Column generation while Daskin et al. (2002) apply Lagrangian relaxation to solve this problem. They use a low-order polynomial algorithm for the nonlinear integer programming sub-problem that must be solved in either of the two approaches. Instead of using the more complex routing costs, they assume linear direct shipment costs from a DC to the customers it serves. They also assume that the variance of demand of customer  $i$ ,  $\sigma_i^2$ , is proportional to the mean demand of customer  $i$ ,  $\mu_i$ , and that the proportionality constant is the same for each customer  $i$ . In other words, they

assume  $\mu_i = \gamma\sigma_i^2 \forall i \in I$ . This allows them to reduce the number of nonlinear terms in the objective function from two to one. Recently, [Shu et al. \(2005\)](#) study a more general model in which this assumption on demand is relaxed.

[Simchi-Levi \(1992\)](#) considers a hierarchical planning model for stochastic distribution systems, in which the locations and demand of customers are determined according to some probability distribution. Different decisions are grouped into three classes: strategic planning, tactical planning, and operational control. He believes that these classes are not independent and an integrated approach is required to avoid sub-optimization, but he did not pursue this approach and instead proposes a hierarchical approach in which the operational costs (location costs and routing costs) of the probabilistic multi-depot distribution system are estimated first, then the service territories for each DC is designed, and finally, routing strategy for the system is determined. The fixed cost of establishing a DC is assumed to be a constant. Also, no inventory related costs are included in his model.

[Vidal and Goetschalckx \(2001\)](#) study a global supply chain design model that maximizes the aftertax profits of a multi-national corporation. Their model simultaneously considering transferring prices, transportation cost allocation, inventory costs, and their impact on the selection of international transportation modes. They propose an algorithm that can produce feasible solutions with small gaps between the solutions and their upper bound. In a recent paper, [Santoso et al. \(2005\)](#) propose a stochastic programming approach for the same problem that is computationally shown to be more efficient when solving large-scale problems. [Alonso-ayuso et al. \(2003\)](#) study a two-stage supply chain planning problem. The first stage deals with the strategic decisions, such as plant sizing, product/material selection and allocation decisions, and the second stage deals with tactical decisions, such as production and inventory levels. The objective is to maximize the expected profit of the supply chain.

In this paper, we propose an integrated stochastic supply chain design model that takes into consideration the location, inventory, and routing costs. We get rid of the assumption that  $\mu_i = \gamma\sigma_i^2 \forall i \in I$  in [Shen et al. \(2003\)](#). Furthermore, we model the shipment from a DC to its customers using a vehicle routing model instead of the linear direct shipping model.

We want to point out that there are other strategic decisions other than location, for instance, capacity, technology, and product mix decisions, that are not included in our model. [Dasci and Verter \(2001a,b\)](#) discuss the technology acquisition and plant location problem, while [Goetschalckx et al. \(2002\)](#) give an excellent review of integrated supply chain design problems.

### 3. Model formulation

In this paper, we assume the customers are uniformly scattered in a connected region,  $\mathcal{A}$ , and the area of  $\mathcal{A}$  is  $A$ . We also assume that the customer demands are independent and follow Normal distributions. The following notation will be used throughout this paper.

#### *Inputs and parameters*

- $I$ : set of customers;
- $J$ : set of candidate DC locations;
- $\mu_i$ : mean (yearly) demand at customer  $i$ , for each  $i \in I$ ;
- $\sigma_i^2$ : variance of (daily) demand at customer  $i$ , for each  $i \in I$ ;
- $f_j$ : fixed (annual) cost of locating a DC at  $j$ , for each  $j \in J$ ;
- $\alpha$ : desired percentage of customers orders satisfied (fill rate);
- $\beta$ : weight factor associated with the shipment cost;
- $\theta$ : weight factor associated with the inventory cost;
- $z_\alpha$ : standard normal deviate such that  $P(z \leq z_\alpha) = \alpha$ ;
- $h$ : inventory holding cost per unit of product per year;
- $F_j$ : fixed administrative and handling cost of placing an order at DC  $j$ , for each  $j \in J$ ;
- $L$ : DC order lead time in days;
- $g_j$ : fixed shipment cost per shipment from the plant to distribution center  $j$ ;
- $\bar{a}_j$ : cost per unit of a shipment from the plant to candidate site  $j$ ;

The weight factors  $\beta$ ,  $\theta$  will be used to adjust the relative proportion of different cost components, as explained in Section 5.

To simplify notation, we assume all the lead times are equal and the holding cost rates are the same at different DCs.

### 3.1. Working inventory cost

In this subsection, we outline the inventory policy under which the system operates. A DC orders inventory from the plant using a  $(r, Q)$  policy with service level constraints. The frequency of orders and the order quantity at each DC are determined by the mean demand served by the DC which, in turn, is a function of the assignment of customers to the DC.

Let  $S_j$  denote the set of customers served by DC  $j$ ,  $D_j$  denote the total annual (expected) demand going through DC  $j$  ( $D_j = \sum_{i \in S_j} \mu_i$ ), and  $n$  be the number of shipments per year from the supplier. Then the average shipment size in one shipment from supplier to DC  $j$  is  $D_j/n$ , and the average working inventory cost at DC  $j$  is  $\theta h D_j / (2n)$ . Assuming the delivery cost from the supplier to DC  $j$  can be calculated as  $g_j + \bar{a}_j D_j / n$ , where  $g_j$  is the fixed cost of placing an order. Then the total annual cost of ordering inventory from the supplier to DC  $j$  is given by

$$F_j n + \beta(g_j + \bar{a}_j D_j / n)n + \theta h D_j / (2n). \quad (1)$$

It is easy to show that the optimal value of  $n$  that minimizes the above function equals to  $\sqrt{\theta h D_j / (2(F_j + \beta g_j))}$ . The corresponding total annual working inventory cost associated with DC  $j$  can be expressed as

$$\sqrt{2\theta h D_j (F_j + \beta g_j)} + \beta \bar{a}_j D_j. \quad (2)$$

### 3.2. Safety stock cost

Since the customer demands are uncorrelated and normally distributed, the total leadtime demand variance seen at DC  $j$  can be written as  $L \sum_{i \in S_j} \sigma_i^2$ . Thus the amount of safety stock required to ensure that stockouts occur with a probability of  $\alpha$  or less is  $z_\alpha \sqrt{L \sum_{i \in S_j} \sigma_i^2}$ . The corresponding holding cost for the safety stock at DC  $j$  is

$$\theta h z_\alpha \sqrt{L \sum_{i \in S_j} \sigma_i^2}. \quad (3)$$

### 3.3. Routing cost approximation

The vehicle routing problem (VRP) is an NP-hard problem, furthermore, since we focus on the design phase and only want to estimate the total expected routing costs as a result of different DC locations, we decide to use continuous approximation to approximate the optimal routing cost. Continuous approximation models, which use continuous functions to represent distributions of customer location and demand, have been developed to provide insights into complicated mathematical programming models. For a review of continuous approximation models and their application in logistics, see Daganzo (1996), Langevin et al. (1996) and Dasci and Verter (2001a,b).

We assume each DC sends a truck to visit its customers at a fixed frequency, say every day or every week. We use  $\chi$  to denote the number of visits in a year. Let  $m$  be the total number of customers served by a specific DC  $j$ ,  $\bar{q}$  be the vehicle capacity,  $d_{ij}$  be the distance between customer  $i$  and DC  $j$ , and  $T_j^*$  be the length of the optimal travelling salesman tour that visits DC  $j$  and the customers it serves. Haimovich and Rinnooy Kan (1985) show that the optimal VRP distance  $V_j$  can be approximated by the following formula:

$$V_j \approx 2 \left( \sum_{i=1}^m \frac{\mu_i}{\chi} d_{ij} \right) / \bar{q} + (1 - 1/\bar{q}) T_j^*. \quad (4)$$

Daganzo (1996) shows that if  $m$  customers are independently scattered in a region according to a spatial customer density (points per unit area)  $\delta(a)$ , and probability density function  $f(a)$  of the customer coordinates  $a = (a_1, a_2)$ , then it is easy to see that  $\delta(a) = mf(a)$ , and the expected tour length  $T_j$  can be expressed as

$$T_j^* \cong \phi m E(\delta(a)^{-1/2}) = \phi m E((mf(a))^{-1/2}) = \phi \sqrt{m} E(f(a)^{-1/2}), \quad (5)$$

where  $\phi$  is an unknown constant.  $\phi = 0.75$  for Euclidean metrics.

Now, let's assume there are  $|I| = N$  customers uniformly scattered in  $\mathcal{A}$ . Among all these  $N$  customers, suppose  $m$  of them are served by DC  $j$  and the other  $N - m$  customers will be served by other DCs. Thus, we can divide  $\mathcal{A}$  into two areas:  $\mathcal{A}_1$  that is occupied by the customers assigned to DC  $j$ , and  $\mathcal{A}_2$  that is occupied by the other  $N - m$  customers. Define a new distribution  $\bar{f}(a)$  for  $\mathcal{A}$  as an estimation of the probability density function (pdf) of these  $m$  customers' locations over  $\mathcal{A}$ :

$$\begin{cases} \bar{f}(a) = \frac{N}{mA}; & a \in \mathcal{A}_1, \\ \bar{f}(a) = 0; & a \in \mathcal{A}_2. \end{cases}$$

Then we can estimate  $T_j^*$  as follows:

$$\begin{aligned} T_j^* &\cong \phi \sqrt{m} E(\bar{f}(a)^{-1/2}) = \phi \sqrt{m} \int_{a \in \mathcal{A}} \sqrt{\bar{f}(a)} da = \phi \sqrt{m} \sum_{i=1,2} \int_{a \in \mathcal{A}_i} \sqrt{\bar{f}(a)} da \cong \phi \sqrt{m} \sum_i |A_i| \sqrt{\bar{f}(a)} \\ &= \phi \sqrt{m} \frac{A}{N} \sqrt{\frac{N}{mA}} = \phi m \sqrt{\frac{A}{N}}. \end{aligned} \quad (6)$$

Substitute (6) into (4), we can estimate the optimal VRP distance by the following formulation:

$$V_j \approx 2 \left( \sum_{i=1}^m \frac{\mu_i}{\chi} d_{ij} \right) / q + (1 - 1/q) \phi m \sqrt{\frac{A}{N}}. \quad (7)$$

From our computational tests, the above approximation is pretty accurate when  $N$  is large enough. We tested the performance of Eq. (7) using a data set with 150 points from Christofides et al. (1979). We first pick one point as the DC and then randomly choose a certain number of points (ranging from 20 to 150) from these 150 points to be the customers served by this DC. We then compare the solutions from Eq. (7) with solutions from a genetic algorithm (GA). From computational study, we believe the GA algorithm can produce near-optimal solutions. We ran the GA algorithm on the 14 test problems described in Christofides et al. (1979). These problems contain between 50 and 199 customers in addition to the depot. The computational results show that the GA algorithm produces solutions that are with 2% of the best solutions that are obtained by meta-heuristics (Agarwal et al., 2004). Fig. 1 depicts the average gaps between the solutions from our approximation algorithm and the genetic algorithm.

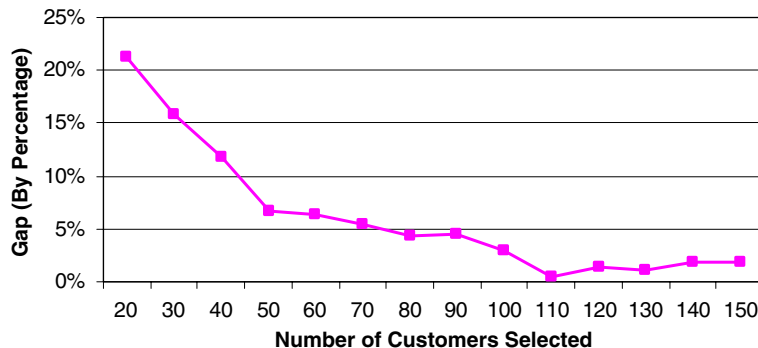


Fig. 1. Comparisons between solutions from Eq. (7) and GA.

As pointed out in Daskin et al. (2002), if the demand satisfy  $\mu_i = \gamma\sigma_i^2 \forall i \in I$ , every DC has a region of service in the optimal solution, and there is no “overlap” of different service areas. Our extensive computational tests show that this property still holds for the case with general demand distributions. We further tested the performance of Eq. (7) under such scenario, where the chosen points are concentrated in a certain area instead of scattering throughout the entire service area (although the shape of the area can be arbitrary). The results show that the gaps are typically smaller than those shown in Fig. 1.

We assume that there is a dedicated truck in each DC that delivers to the retailers every period according to a certain route. It is reasonable to assume that, under some conditions (e.g., driver does not have to work over-time), the transportation cost related to this route is concave in the distance travelled. That is, the routing cost  $RC_j(\cdot)$  should be a concave function of  $V_j$ .

### 3.4. Integrated model

We define the following *Decision variables*:

- $X_j = 1$ , if  $j$  is selected as a DC location, and 0, otherwise, for each  $j \in J$ ;
- $Y_{ij} = 1$ , if customer  $i$  is serviced by a DC based at  $j$ , and 0, otherwise, for each  $i \in I$  and each  $j \in J$ .

As we show in Section 3.3, we can estimate the routing distance from DC  $j$  to the customers it serves by the following formulation:

$$V_j \approx 2 \left( \sum_{i \in I} \mu_i d_{ij} Y_{ij} \right) / \bar{q} + \chi(1 - 1/\bar{q})\phi \sum_{i \in I} Y_{ij} \sqrt{\frac{A}{N}} = \sum_{i \in I} \hat{b}_{ij} Y_{ij}, \quad (8)$$

where  $\hat{b}_{ij} = 2\mu_i d_{ij}/\bar{q} + \chi(1 - 1/\bar{q})\phi\sqrt{\frac{A}{N}} \geq 0$ .

By incorporating (2) and (3), the problem can be formulated as

$$\text{Problem } P: \quad \min \quad \sum_{j \in J} \left\{ f_j X_j + \beta \bar{a}_j \sum_{i \in I} \mu_i Y_{ij} + RC_j \left( \sum_{i \in I} \hat{b}_{ij} Y_{ij} \right) + K_j \sqrt{\sum_{i \in I} \mu_i Y_{ij}} + q \sqrt{\sum_{i \in I} \sigma_i^2 Y_{ij}} \right\} \quad (9)$$

$$\text{s.t.} \quad \sum_{j \in J} Y_{ij} = 1 \quad \text{for each } i \in I, \quad (10)$$

$$Y_{ij} - X_j \leq 0 \quad \text{for each } i \in I, \quad j \in J, \quad (11)$$

$$Y_{ij} \in \{0, 1\} \quad \text{for each } i \in I, \quad j \in J, \quad (12)$$

$$X_j \in \{0, 1\} \quad \text{for each } j \in J, \quad (13)$$

where

$$K_j = \sqrt{2\theta(F_j + \beta g_j)h},$$

$$q = \theta h z_\alpha \sqrt{L}.$$

Since  $K_j\sqrt{x}$  denotes the working inventory cost, and  $q\sqrt{x}$  represents the safety inventory cost, we define

$$\text{WIC}_j(x) := K_j\sqrt{x},$$

$$\text{SIC}_j(x) := q\sqrt{x}$$

and rewrite the model objective as

$$\min \quad \sum_{j \in J} \left( f_j X_j + \beta \bar{a}_j \sum_{i \in I} \mu_i Y_{ij} + RC_j \left( \sum_{i \in I} \hat{b}_{ij} Y_{ij} \right) + \text{WIC}_j \left( \sum_{i \in I} \mu_i Y_{ij} \right) + \text{SIC}_j \left( \sum_{i \in S} \sigma_i^2 Y_{ij} \right) \right). \quad (14)$$

The objective function minimizes the weighted sum of the following four cost components:

- The fixed cost of locating facilities, given by the term  $\sum_j f_j X_j$ .
- The annual shipment cost from the supplier to the DCs, given by the term  $\sum_j \beta \bar{a}_j \sum_{i \in I} \mu_i Y_{ij}$ .
- The annual shipment cost from the DCs to the customers, given by the term  $\sum_j RC_j (\sum_{i \in I} \hat{b}_{ij} Y_{ij})$ .
- The expected total inventory costs, given by  $\sum_j K_j \sqrt{\sum_{i \in I} \mu_i Y_{ij}} + \sum_j q \sqrt{\sum_{i \in I} \sigma_i^2 Y_{ij}}$ .

The constraints of the model are identical to those of the uncapacitated facility location (UFL) problem, thus the problem we are studying is more difficult than the standard UFL problem, which is already a notorious NP-hard problem.

#### 4. Solution approach

We use Lagrangian relaxation embedded in branch and bound to solve Problem  $P$ . In the following we explain how to derive the lower bound and upper bound of Problem  $P$ . We also present a variable fixing technique to speed up the algorithm. Daskin et al. (2002) use a similar Lagrangian approach to solve a special case of our general model, where there is only one separable concave term. For the same optimization problem, Verter and Dincer (1995) and Dasci and Verter (2001a,b) propose the progressive piecewise linear underestimation technique. No algorithm has been reported in the literature that can solve a model with three or more separable concave terms in the objective function.

##### 4.1. Finding a lower bound

We relax constraint (10) to obtain the following Lagrangian Dual problem:

$$\begin{aligned}
 \max_{\lambda} \min_{X, Y} \quad & \sum_{j \in J} \left\{ f_j X_j + \beta \bar{a}_j \sum_{i \in I} \mu_i Y_{ij} + RC_j \left( \sum_{i \in I} \hat{b}_{ij} Y_{ij} \right) + WIC_j \left( \sum_{i \in I} \mu_i Y_{ij} \right) \right. \\
 & \left. + SIC_j \left( \sum_{i \in I} \sigma_i^2 Y_{ij} \right) \right\} + \sum_{i \in I} \lambda_i \left( 1 - \sum_{j \in J} Y_{ij} \right) \\
 = \sum_{j \in J} & \left[ f_j X_j + \sum_{i \in I} (\beta \bar{a}_j \mu_i - \lambda_i) Y_{ij} + RC_j \left( \sum_{i \in I} \hat{b}_{ij} Y_{ij} \right) \right. \\
 & \left. + WIC_j \left( \sum_{i \in I} \mu_i Y_{ij} \right) + SIC_j \left( \sum_{i \in I} \sigma_i^2 Y_{ij} \right) \right] + \sum_{i \in I} \lambda_i \\
 \text{s.t.} \quad & Y_{ij} - X_j \leq 0 \quad \text{for each } i \in I, j \in J, \\
 & Y_{ij} \in \{0, 1\} \quad \text{for each } i \in I, j \in J, \\
 & X_j \in \{0, 1\} \quad \text{for each } j \in J.
 \end{aligned} \tag{15}$$

For fixed values of the Lagrange multiplier,  $\lambda_i$ , we want to minimize (15) over the location variables,  $X_j$ , and the assignment variables,  $Y_{ij}$ . In order to solve this problem, we need to be able to solve a sub-problem  $P(j)$  of the following form for each candidate DC  $j \in J$ :

$$\begin{aligned}
 \text{Problem } P(j) : \tilde{V}_j = \min \quad & \sum_{i \in I} a_i Y_{ij} + RC_j \left( \sum_{i \in I} b_i Y_{ij} \right) + WIC_j \left( \sum_{i \in I} c_i Y_{ij} \right) + SIC_j \left( \sum_{i \in I} d_i Y_{ij} \right) \\
 & Y_{ij} \in \{0, 1\} \quad \text{for all } i \in I,
 \end{aligned} \tag{16}$$

where

$$\begin{aligned}
 a_i &:= \beta \bar{a}_j \mu_i - \lambda_i, \\
 b_i &:= \hat{b}_{ij},
 \end{aligned}$$



$$\begin{aligned} c_i &:= \mu_i, \\ d_i &:= \sigma_i^2. \end{aligned}$$

Note that  $f_j$  does not depend on  $Y_{ij}$  and hence can be ignored for discussion here.

Define  $r_j \equiv \tilde{V}_j + f_j$ , then it is easy to see that  $\text{LB} \equiv \sum_{j:r_j \leq 0} r_j + \sum_i \lambda_i$  is a lower bound to Problem  $P$ .

For each  $j \in J$ , define set function  $G_j$  on  $I$  as follows: For each  $S \subseteq I$ ,

$$G_j(S) \equiv \sum_{i \in S} a_i + \text{RC}_j \left( \sum_{i \in S} b_i \right) + \text{WIC}_j \left( \sum_{i \in S} c_i \right) + \text{SIC}_j \left( \sum_{i \in S} d_i \right). \quad (17)$$

#### 4.1.1. Properties of the problem

We show in this section some properties of Problem  $P(j)$ . We use  $R_j^*$  to denote the optimal solution to  $\min_{S \subseteq I} G_j(S)$ . We first show that instead of checking every customer in set  $I$ , we can just focus on a subset of  $I$ .

**Lemma 1.** *Given a candidate DC location  $j \in J$ , and associated minimum-reduced-cost set  $R_j^* \subset I$ . For every  $i \in R_j^*$ ,  $a_i < 0$ .*

**Proof.** Let  $i \in R_j^*$ . Since  $b_i, c_i, d_i > 0$ , if  $a_i \geq 0$ , then for any solution  $\bar{z}$  with  $\bar{z}_i = 1$ , the objective function value is strictly greater than that of the solution obtained from  $\bar{y}$  by setting  $\bar{z}_i = 0$ .  $\square$

Hence we may restrict our search for  $R_j^*$  to customers in  $\Gamma$ , where  $\Gamma \equiv \{i \in I : a_i < 0\}$ . We next identify a nice structural property of set  $R_j^*$  by extending an argument in Chakravarty et al. (1985) and Shu et al. (2005).

Let  $a_S = \sum_{i \in S} a_i$ ,  $b_S = \sum_{i \in S} b_i$ ,  $c_S = \sum_{i \in S} c_i$ , and  $d_S = \sum_{i \in S} d_i$ . Define a new function

$$h_j(x_1, x_2, x_3, x_4) := x_1 + \text{RC}_j(x_2) + \text{WIC}_j(x_3) + \text{SIC}_j(x_4). \quad (18)$$

Note that  $h_j(x_1, x_2, x_3, x_4)$  is a separable concave function. It is clear that

$$\min_{S \subseteq \Gamma} G_j(S) = \min_{S \subseteq \Gamma} h_j(a_S, b_S, c_S, d_S). \quad (19)$$

Since the set  $\{(a_S, b_S, c_S, d_S) : S \subseteq \Gamma\}$  is finite, its convex hull, which will be denoted by  $H$ , is a convex polyhedron.

$$\min_{S \subseteq \Gamma} G_j(S) = \min_{S \subseteq \Gamma} h_j(a_S, b_S, c_S, d_S) = \min_{(a,b,c,d) \in H} h_j(a, b, c, d),$$

the latter minimization problem attains a minimum at an extreme point of  $H$  since the function  $h_j$  is concave.

Let  $(a^*, b^*, c^*, d^*)$  be an extreme point of  $H$ . Since  $H$  is a polyhedron, it is well known that there exists a linear function  $f(\cdot)$  on  $H$  that attains its unique minimum over  $H$  at  $(a^*, b^*, c^*, d^*)$ . Since  $f$  is linear, it has a representation  $f(a, b, c, d) = \alpha'a + \beta'b + \gamma'c + \delta'd$  defined by real numbers  $\alpha'$ ,  $\beta'$ ,  $\gamma'$  and  $\delta'$ . The uniqueness of  $(a^*, b^*, c^*, d^*)$  as the minimizer of  $f$  over  $H$  assures that we do not have  $\alpha' = \beta' = \gamma' = \delta' = 0$ .

Since  $H$  is the convex hull of  $\{(a_S, b_S, c_S, d_S) : S \subseteq \Gamma\}$ ,

$$\begin{aligned} \alpha'a^* + \beta'b^* + \gamma'c^* + \delta'd^* &= \min_{(a,b,c,d) \in H} \alpha'a + \beta'b + \gamma'c + \delta'd = \min_{S \subseteq \Gamma} \alpha'a_S + \beta'b_S + \gamma'c_S + \delta'd_S \\ &= \min_{S \subseteq \Gamma} \sum_{i \in S} (\alpha'a_i + \beta'b_i + \gamma'c_i + \delta'd_i). \end{aligned}$$

The set  $S^* = \{i \in \Gamma : \alpha'a_i + \beta'b_i + \gamma'c_i + \delta'd_i < 0\}$  is clearly optimal for the above optimization problem. Hence, we conclude from the uniqueness property that  $(a^*, b^*, c^*, d^*) = (a_{S^*}, b_{S^*}, c_{S^*}, d_{S^*})$ , i.e.,  $R_j^* = S^*$ . Note that  $S^* = \{i : \alpha'a_i + \beta'b_i + \gamma'c_i + \delta'd_i < 0\} = \{i : \alpha' + \beta' \frac{b_i}{a_i} + \gamma' \frac{c_i}{a_i} + \delta' \frac{d_i}{a_i} > 0\} = \{i : \beta'x_i + \gamma'y_i + \delta'z_i < \alpha'\}$ , where  $x_i = -b_i/a_i$ ,  $y_i = -c_i/a_i$ , and  $z_i = -d_i/a_i$ . Here,  $x_i, y_i, z_i \geq 0$  for all  $i$ .

If we define a point  $p_i$  for each  $i \in \Gamma$ , with coordinate  $(x_i, y_i, z_i)$  in the three-dimension space  $OXYZ$ , and denote the set of point  $\{p_i, i \in \Gamma\}$  by  $R$ , then  $S^*$  includes all points in  $R$  that locate below the plane defined by  $\beta'x_i + \gamma'y_i + \delta'z_i = \alpha'$ , i.e., all  $p_i \in R$  that satisfy  $\beta'x_i + \gamma'y_i + \delta'z_i < \alpha'$ . However, we do not know the exact values of  $\alpha'$ ,  $\beta'$ ,  $\gamma'$  and  $\delta'$ , thus we can not derive  $S^*$  directly. In order to determine  $S^*$ , and hence the optimal solution to Problem  $P(j)$ , our strategy is to enumerate all possible planes defined by different values of  $\alpha'$ ,



$\beta'$ ,  $\gamma'$  and  $\delta'$  that result in different partitions of set  $R$ , thus obtaining all possible candidate solutions for set  $S^*$ . We then choose the best one based on the corresponding objective values of Problem  $P(j)$ . We show how this process works in the following section.

#### 4.1.2. Rank and search algorithm

We first observe that any two points in  $R$ , say  $p_1$  and  $p_2$ , together with another point, say  $p_r \in R$ ,  $r \geq 3$ , that is linearly independent with  $p_1$  and  $p_2$ , can determine a plane,  $F_r(x, y, z) = 0$ . We also create a unique base plane  $F_b$ , containing  $p_1$ ,  $p_2$  and vertical with the  $z = 0$  plane. Let the plane that is vertical with  $F_b$  and contains both  $p_1$  and  $p_2$  be  $F_v$ . Plane  $F_r(x, y, z) = 0$  partitions set  $R$  into three subsets. See Fig. 2 for graphical illustration.

- **Region<sub>A</sub>**:  $\{p_i : F_r(x_i, y_i, z_i) > 0, i \in \Gamma\}$ ;
- **Region<sub>B</sub>**:  $\{p_i : F_r(x_i, y_i, z_i) < 0, i \in \Gamma\}$ ;
- $F_r$ :  $\{p_i : F_r(x_i, y_i, z_i) = 0, i \in \Gamma\}$ . Furthermore,  $p_1$  and  $p_2$  determine a certain line in the plane  $F_r(x_i, y_i, z_i) = 0$ , say  $ax + by = c$ , which partitions  $F_r$  into seven disjoint subsets:
  - $SR_1$ :  $F_r \cap \{p_i : ax_i + by_i > c, i \in \Gamma\}$ ;
  - $SR_2$ :  $F_r \cap \{p_i : ax_i + by_i < c, i \in \Gamma\}$ ;
  - $I_1$ :  $\{p_i : ax_i + by_i = c \text{ and } x_i < x_1, i \in \Gamma\}$ ;
  - $I_2$ :  $\{p_i : ax_i + by_i = c \text{ and } x_1 < x_i < x_2, i \in \Gamma\}$ ;
  - $I_3$ :  $\{p_i : ax_i + by_i = c \text{ and } x_2 < x_i, i \in \Gamma\}$ ;
  - $P_1$ :  $\{p_i : \text{with the same location as } p_1, i \in \Gamma\}$ ;
  - $P_2$ :  $\{p_i : \text{with the same location as } p_2, i \in \Gamma\}$ .

We define set

$$CS(p_1, p_2) = \left\{ SR_1, SR_1 \cup I_1, SR_1 \cup I_1 \cup P_1, SR_1 \cup I_1 \cup P_1 \cup I_2, SR_1 \cup I_1 \cup P_1 \cup I_2 \cup P_2, \right. \\ SR_1 \cup I_1 \cup P_1 \cup I_2 \cup P_2 \cup I_3, SR_1, SR_1 \cup I_3, SR_1 \cup I_3 \cup P_2, SR_1 \cup I_3 \cup P_2 \cup I_2, \\ SR_1 \cup I_3 \cup P_2 \cup I_2 \cup P_1, SR_2, SR_2 \cup I_1, SR_2 \cup I_1 \cup P_1, SR_2 \cup I_1 \cup P_1 \cup I_2, \\ SR_2 \cup I_1 \cup P_1 \cup I_2 \cup P_2, SR_2 \cup I_1 \cup P_1 \cup I_2 \cup P_2 \cup I_3, SR_2, SR_2 \cup I_3, \\ \left. SR_2 \cup I_3 \cup P_2, SR_2 \cup I_3 \cup P_2 \cup I_2, SR_2 \cup I_3 \cup P_2 \cup I_2 \cup P_1 \right\}.$$

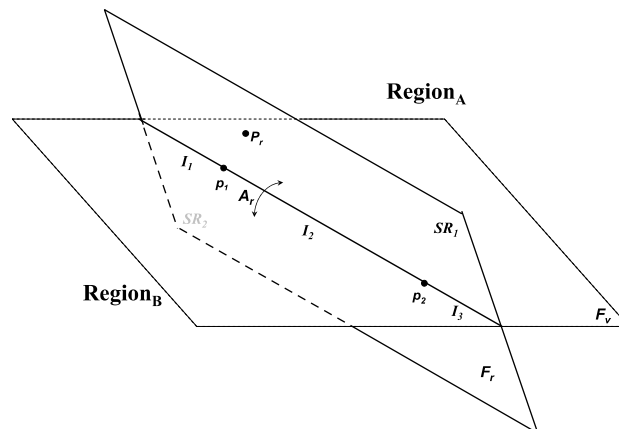


Fig. 2. 3-D Illustration of the algorithm.

Based on the regions defined above, it is easy to see  $S^*$  corresponds to one of the following subset of  $R$ :

- Region<sub>A</sub>;
- Region<sub>A</sub>  $\cup E$  (any  $E \in CS(p_1, p_2)$ );
- Region<sub>B</sub>;
- Region<sub>B</sub>  $\cup E$  (any  $E \in CS(p_1, p_2)$ ).

The following *Rank and Search* algorithm is designed to enumerate the above subsets efficiently.

First, we determine a certain sequence for each pair of points in  $R$ . For instance, the sequence can go in the following order:  $p_1, p_2; p_1, p_3; \dots; p_1, p_n; \dots; p_{n-1}, p_n$ . Each pair will be processed in one iteration. Without loss of generality, assume in the current iteration, we are checking pair  $p_1, p_2$ , which we call the *current checking pair*. We use  $(x_i, y_i, z_i)$  to denote  $p_i$ 's coordinates. In the following, we discuss the algorithm according to three cases.

**Case 1.** If  $x_1 \neq x_2$  or  $y_1 \neq y_2$  (please refer to Fig. 3 for illustration)

1. Plane  $F_b$  divides the three-dimension space into two subspaces,  $S_1$  (including  $F_b$ ) and  $S_2$ . It also divides Set  $R$  into  $R_{S1}$  and  $R_{S2}$ , where  $R_{S1} \subseteq R$  contains all points in  $R$  that are located within  $S_1$ , and  $R_{S2} = R \setminus R_{S1}$ .
2. For any other point, say  $p_r$  ( $r \geq 3$ ):
  - If  $p_r$  is not on the line created by  $p_1$  and  $p_2$ , we can get a plane  $F_r$  that contains  $p_1, p_2$  and  $p_r$ . Compute the angle (the one less than or equal to  $90^\circ$ ),  $A_r$ , between plane  $F_r$  and  $F_v$ . If  $p_r \in R_{S1}$  is below the plane  $F_v$  or  $p_r \in R_{S2}$  is above the plane  $F_v$ , let  $A_r = -A_r$ .
  - If  $p_r$  is on the line created by  $p_1$  and  $p_2$ , let  $A_r = -100^\circ$ .
3. Rank the points in  $R_{S1}$  (and also  $R_{S2}$ ) non-increasingly according to the angle  $A_r$ .
4. For each point  $p_r$ , say  $p_r \in R_{S1}$ , the subsets determined by the current checking pair and  $p_r$  are as follows (similar results can be derived for each point in  $R_{S2}$ ):
  - Region<sub>A</sub> =  $\{j : p_j \in R_{S1} \text{ and } A_j > A_r\} \cup \{j : p_j \in R_{S2} \text{ and } A_j < A_r\}$ ;
  - Region<sub>B</sub> =  $\{j : p_j \in R_{S1} \text{ and } -90^\circ \leq A_j < A_r\} \cup \{j : p_j \in R_{S2} \text{ and } A_j > A_r\}$ ;
  - $SR_1 = \{j : p_j \in R_{S1} \text{ and } A_j = A_r\}$ ;
  - $SR_2 = \{j : p_j \in R_{S2} \text{ and } A_j = A_r\}$ ;
  - If  $A_j = -100^\circ$ , then  $j$  is a candidate for the sets  $I_1, I_2, I_3, P_1$  or  $P_2$  depending on its relative location with respect to  $p_1$  and  $p_2$ .

The most time consuming step is Step 3, which takes  $O(N \log N)$ . After that, each candidate solution (for each successive choice of  $r$ ) can be constructed in  $O(1)$  time.

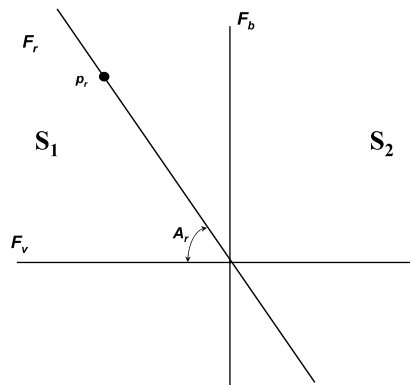


Fig. 3. 2-D Illustration of the algorithm.

**Case 2.** If  $x_1 = x_2$ ,  $y_1 = y_2$  and  $z_1 \neq z_2$

Then  $p_1$  and  $p_2$  together with any other point in  $R$  can determine a plane that either contains or is parallel with the  $z = 0$  axis. This means we can reduce the problem to two-dimension ( $OXY$ ), which has been studied in Shu et al. (2005).

**Case 3.** If  $x_1 = x_2$ ,  $y_1 = y_2$  and  $z_1 = z_2$

We can skip this iteration since  $p_2$  is located at the same position as  $p_1$ .

It is easy to see that in order to enumerate all possible solutions for  $S^*$ , and choose the best one among them to solve Problem  $P(j)$ , we need  $O(N^2)$  iterations and each iteration takes  $O(N \log N)$ , thus the complexity of this algorithm is  $O(N^3 \log N)$ .

#### 4.2. Finding an upper bound

At each iteration of the Lagrangian procedure, we find an upper bound as follows. We initially fix the DC locations at those sites for which  $X_j = 1$  in the current Lagrangian solution. Then we assign customers to DCs according to the following two cases.

- For each customer assigned to at least one open DC in the Lagrangian solution, we assign the customer to the DC  $j$  for which  $Y_{ij} = 1$  and that increases the cost the least based on the assignments made so far.
- For customers that were not assigned to any open DC in the Lagrangian solution, we assign each such customer to the open DC which increases the total cost the least based on the assignments made so far.

#### 4.3. Variable fixing

In the straightforward implementation of the above algorithm, we need to solve, for each  $j \in J$ , problem  $P(j)$  for each Lagrangian iteration. It will reduce the solution times dramatically if we can identify some DCs what will not be included in the optimal solution and thus we do not have to solve problem  $P(j)$  associated with  $j$ . We show next how information on the upper bound and lower bound can be used to achieve the above goal.

We use  $Z_{IP}$  to denote the value of the optimal integral solution to Problem  $P$ . Let  $j^*$  be a facility such that  $r_{j^*} > 0$ , and let UB be an upper bound for Problem  $P$ .

**Claim 1.** If  $LB + r_{j^*} > UB$ , then  $j^*$  will never be used as a DC in the optimal solution to Problem  $P$ .

This is easy to show by contradiction. Suppose  $j^*$  is included in the solution, then  $Z_{IP}$  remains unchanged if we impose the additional condition:  $X_{j^*} = 1$  to the existing set of constraints. Thus,  $Z_{IP} \geq \sum_{j:r_j \leq 0} r_j + \sum_{i,l} \lambda_{il} + r_{j^*} \equiv LB + r_{j^*}$ . On the other hand,  $Z_{IP} \leq UB$ , which gives rise to a contradiction.

Note that once we determine that  $j^*$  will never be used as a DC in the optimal solution, we do not need to solve the corresponding pricing problem anymore in the rest of the Lagrangian relaxation procedure.

Finally, if after the variable forcing routine, either the lower bound equals the upper bound or there are no unforced DC location variables, then the solution corresponding to the upper bound is optimal, the lower bound is then set to the upper bound and the algorithm terminates. Our computational results did not find any instances in which the problem could not be solved to optimality at the root node of the branch and bound tree, so we have not implemented a full branch and price code. We can easily implement such a code by following the branching rules in Daskin et al. (2002).

### 5. Computation results

In this section, we present computational results for four different data sets, with sizes ranging from 40 to 320 customers. In all the data sets, we assume the customers are geographically dispersed in a square region. We choose DCs from the locations of customers.

We use the following parameter values:

$h = 1$ ;  $F_j = 10$ ;  $g_j = 10$ ;  $a_j = 5$ ;  $L = 7$ ;  
 $z_\alpha = 1.96$  (97.5% service level);  
 $\mu_i$ : Uniformly drawn from  $[100, 1600]$ ;  
 $\sigma_i^2$ : Uniformly drawn from  $[10, 50]$ ;  
 $f_j$ : Uniformly drawn from  $[100, 200]$ .

For the routing cost  $RC_j$ , which is a concave function of the total distance travelled, we use  $RC_j = \beta\chi(1.4 * V_j + \sqrt{V_j})$  in our tests.  $\chi$  is set to be 250.

We generate instances by varying the values of  $\beta$  (the distribution cost factor) and  $\theta$  (the inventory holding cost factor). Our goal is to find out (1) how solution difficulty and the optimal number of DCs vary with the changing of  $\beta$  and  $\theta$  values, and (2) what is the benefit of integrating inventory and routing decisions in the strategic supply chain design models.

### 5.1. Performance of the algorithm

The algorithm is implemented in C++. All the computational times are obtained on a DELL PC P4 running at 2.8 GHz using Windows XP.

Before we discuss the computational results, we want to point out that in our problem, all the three concave functions are continuously increasing functions, thus we know that  $\beta'$ ,  $\gamma'$  and  $\delta'$  are all non-negative, which implies that in Step 4 of the Rank and Search algorithm, we only need to check points that determine a plane with non-negative norm. Doing so can greatly improve the efficiency of the *rank and search* algorithm.

The parameters for the Lagrangian procedure (Camerini et al., 1975) are shown in Table 1. Table 2 shows the minimal, maximal and average CPU times for different sizes of the problem, and the CPU times reported are based on 10 runs.

### 5.2. Impacts of inventory cost and transportation cost on optimal decision

Tables 3–6 show the computational results for 40, 80, 160 and 320 customers, respectively. By varying the values of  $\beta$  and  $\theta$ , we can see clearly the impacts of transportation costs and inventory costs on DC selection, customer assignment, and total running time of algorithm.

Table 1  
Parameters for Lagrangian relaxation procedure

| Parameter  | Value  |
|--|--|
| Initial $\lambda_i$                                | $\bar{\mu}_i \times \beta \times 10 + f_i$                     |
| Initial scalar                                     | 1.06   |
| Maximum number of iterations before halving scalar | 12   |
| Stopping criterion for Lagrangian relaxation       | Upper Bound – Lower Bound < 0.00001 or all customers are fixed |

Table 2  
Solutions times of the algorithm

| Number of customers | CPU times (second) |         |         |
|---------------------|--------------------|---------|---------|
|                     | Min                | Max     | Average |
| 40                  | 0.119              | 0.340   | 0.208   |
| 80                  | 1.086              | 3.207   | 1.865   |
| 160                 | 13.764             | 27.062  | 21.512  |
| 320                 | 190.318            | 452.633 | 262.851 |

Table 3  
Computation results for the 40-customer problem

| Input   |          | Output     |          |             |             |                     |        |      |       |
|---------|----------|------------|----------|-------------|-------------|---------------------|--------|------|-------|
| $\beta$ | $\theta$ | No. of DCs | CPU Time | Upper bound | Lower bound | Percentage of costs |        |      |       |
|         |          |            |          |             |             | Loc.                | Trans. | Inv. | Order |
| 0.0010  | 0.3      | 8          | 0.296    | 5722.299    | 5722.299    | 20                  | 57     | 12   | 11    |
| 0.0012  | 0.3      | 9          | 0.281    | 6363.871    | 6363.871    | 21                  | 58     | 11   | 10    |
| 0.0014  | 0.3      | 13         | 0.234    | 6933.207    | 6933.207    | 26                  | 51     | 12   | 11    |
| 0.0014  | 0.3      | 13         | 0.234    | 6933.207    | 6933.207    | 26                  | 51     | 12   | 11    |
| 0.0014  | 0.35     | 13         | 0.265    | 7068.915    | 7068.359    | 26                  | 49     | 13   | 12    |
| 0.0014  | 0.4      | 10         | 0.203    | 7188.411    | 7187.938    | 20                  | 57     | 12   | 11    |

Table 4  
Computation results for the 80-customer problem

| Input   |          | Output     |          |             |             |                     |        |      |       |
|---------|----------|------------|----------|-------------|-------------|---------------------|--------|------|-------|
| $\beta$ | $\theta$ | No. of DCs | CPU time | Upper bound | Lower bound | Percentage of costs |        |      |       |
|         |          |            |          |             |             | Loc.                | Trans. | Inv. | Order |
| 0.0010  | 0.4      | 17         | 2.468    | 13130.076   | 13130.009   | 18                  | 59     | 12   | 11    |
| 0.0011  | 0.4      | 24         | 2.828    | 13847.184   | 13847.184   | 23                  | 52     | 13   | 12    |
| 0.0012  | 0.4      | 29         | 2.110    | 14444.516   | 14444.516   | 28                  | 45     | 14   | 13    |
| 0.0012  | 0.4      | 29         | 2.110    | 14444.516   | 14444.516   | 28                  | 45     | 14   | 13    |
| 0.0012  | 0.5      | 27         | 3.828    | 14906.909   | 14906.909   | 25                  | 48     | 14   | 13    |
| 0.0012  | 0.6      | 25         | 2.843    | 15313.350   | 15313.344   | 22                  | 50     | 15   | 13    |

Table 5  
Computation results for the 160-customer problem

| Input   |          | Output     |          |             |             |                     |        |      |       |
|---------|----------|------------|----------|-------------|-------------|---------------------|--------|------|-------|
| $\beta$ | $\theta$ | No. of DCs | CPU time | Upper bound | Lower bound | Percentage of costs |        |      |       |
|         |          |            |          |             |             | Loc.                | Trans. | Inv. | Order |
| 0.0020  | 0.2      | 36         | 21.500   | 23007.965   | 23007.910   | 22                  | 60     | 9    | 9     |
| 0.0022  | 0.2      | 41         | 37.156   | 24326.274   | 24326.272   | 23                  | 59     | 9    | 9     |
| 0.0024  | 0.2      | 45         | 16.921   | 25566.570   | 25566.567   | 24                  | 58     | 9    | 9     |
| 0.0020  | 0.1      | 41         | 41.172   | 21709.354   | 21709.354   | 26                  | 59     | 8    | 7     |
| 0.0020  | 0.2      | 36         | 21.500   | 23007.965   | 23007.910   | 22                  | 60     | 9    | 9     |
| 0.0020  | 0.3      | 29         | 25.812   | 23910.280   | 23910.249   | 17                  | 64     | 10   | 9     |

Table 6  
Computation results for the 320-customer problem

| Input   |          | Output     |          |             |             |                     |        |      |       |
|---------|----------|------------|----------|-------------|-------------|---------------------|--------|------|-------|
| $\beta$ | $\theta$ | No. of DCs | CPU time | Upper bound | Lower bound | Percentage of costs |        |      |       |
|         |          |            |          |             |             | Loc.                | Trans. | Inv. | Order |
| 0.0025  | 0.2      | 53         | 185.516  | 45806.728   | 45806.631   | 16                  | 68     | 8    | 8     |
| 0.0030  | 0.2      | 78         | 138.421  | 51591.010   | 51591.010   | 20                  | 63     | 9    | 8     |
| 0.0035  | 0.2      | 105        | 145.625  | 56553.604   | 56553.604   | 26                  | 56     | 9    | 9     |
| 0.0035  | 0.2      | 105        | 145.625  | 56553.604   | 56553.604   | 26                  | 56     | 9    | 9     |
| 0.0035  | 0.3      | 94         | 128.516  | 58873.031   | 58873.029   | 22                  | 58     | 10   | 10    |
| 0.0035  | 0.4      | 84         | 85.047   | 60725.502   | 60725.480   | 19                  | 60     | 11   | 10    |

We observe from the computational results that as the relative weight of inventory costs goes up, the number of DCs located goes down; and as the importance of transport costs goes up, the number of DCs increases. This observation is consistent with results from the literature (e.g., Shen et al. (2003)).

### 5.3. Benefits of supply chain integration

Ozsen (2004) shows that cost saving can be obtained by considering location and inventory decisions simultaneously instead of the sequential approach, where location decision is made before the inventory decision. She also finds that the larger the proportion of inventory cost, the more the cost saving achieved by integration. She uses direct shipping to model the delivery from DCs to customers in both the integrated and the sequential approaches.

In this section, we study the benefit of integration with one more level of decision—the routing decision. We compare the costs from the following three approaches:

- *Fully integrated approach*: integrate location, inventory, and vehicle routing decisions in the same model. The routing distance is calculated using Eq. (7).
- *Partially integrated approach*: integrate location and inventory decisions with direct shipping assumption to determine the DC locations and customer assignments, then estimate the vehicle routing distance using (7).
- *Sequential approach*: determine the locations of DCs based on uncapacitated facility location model, then make inventory and routing decisions as discussed in Sections 3.1 and 3.3.

All computational experiments are conducted with a given data set with 80 customers. Figs. 4–9 illustrate the percentages of the four cost components for different  $\beta$  and  $\theta$  values. Specifically, Figs. 4 and 5 present the

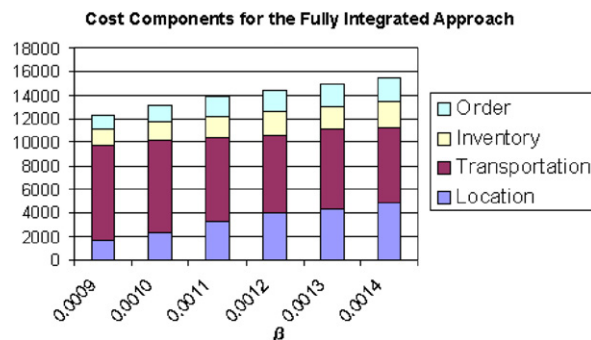


Fig. 4. Cost components for the fully integrated approach as  $\beta$  varies ( $\theta = 0.4$ ).

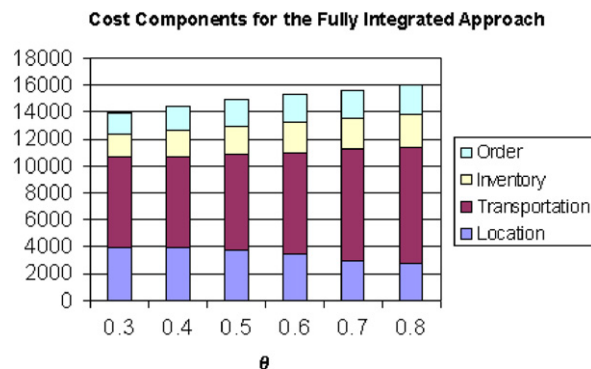


Fig. 5. Cost components for the fully integrated approach as  $\theta$  varies ( $\beta = 0.0012$ ).

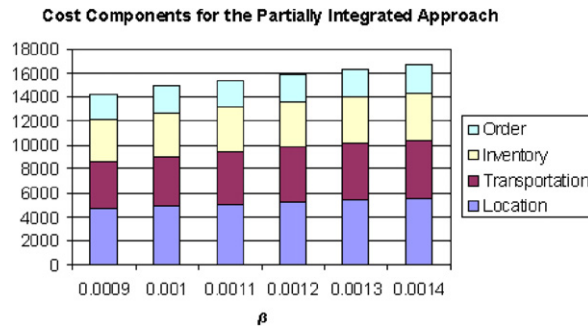


Fig. 6. Cost components for the partially integrated approach as  $\beta$  varies ( $\theta = 0.4$ ).

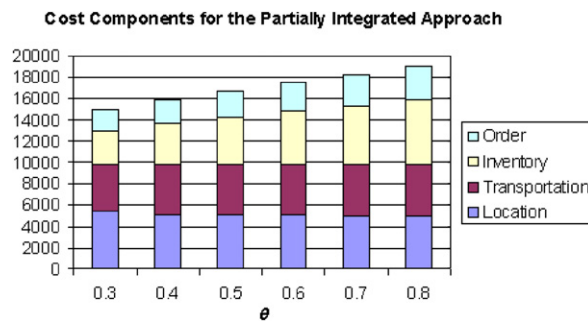


Fig. 7. Cost components for the partially integrated approach as  $\theta$  varies ( $\beta = 0.0012$ ).

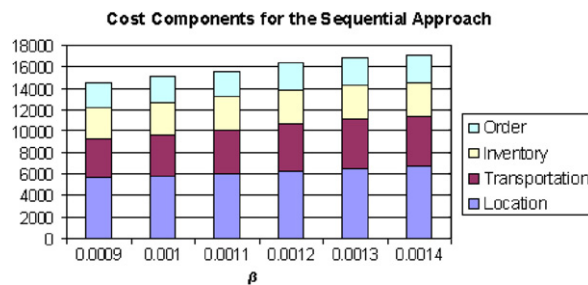


Fig. 8. Cost components for the sequential approach as  $\beta$  varies ( $\theta = 0.4$ ).

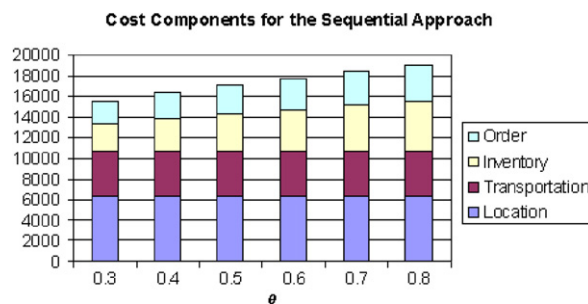


Fig. 9. Cost components for the sequential approach as  $\theta$  varies ( $\beta = 0.0012$ ).

results for the fully integrated approach, Figs. 6 and 7 show those for the partially integrated approach, and Figs. 8 and 9 for the sequential approach.



From these figures, we observe that as the degree of integration increases, fewer DCs would be opened, and the proportion of location cost decreases.

Tables 7 and 8 show the benefit of integration. In these two tables, data in the second and fourth columns, “proportion of the location costs” and “proportion of the location, inventory and order costs”, are calculated for the fully integrated approach.

Comparing with the fully integrated approach, we observe that:

- for the sequential approach, the lower the proportion of location cost in total cost, the more benefit we can obtain by integration;
- for the partially integrated approach, the lower the proportion of location and inventory cost in total cost, the more benefit we can obtain by integration.

Both observations suggest that if the costs related to the tactical and operational decisions account for a large portion of the total cost, then it is more beneficial to take these decisions into consideration when making strategic location decisions.

Table 7  
The benefit of fully integrated approach ( $\theta = 0.4$ )

| $\beta$ | Proportion of the location costs (%) | The percentage improved by the fully integrated approach compared to the sequential approach (%) | Proportion of the location, inventory and order costs (%) | The percentage improved by the fully integrated approach compared to the partially integrated approach (%) |
|---------|--------------------------------------|--|---|--|
| 0.0009  | 13.8593236                           | 18.73125   | 34.153215   | 15.5610879   |
| 0.001   | 17.5627314                           | 14.8672681   | 39.8085706  | 13.2789416   |
| 0.0011  | 23.4199249                           | 12.8498062   | 48.3200327  | 10.9407125   |
| 0.0012  | 27.8652472                           | 12.5872676   | 54.1591473  | 9.8267987  |
| 0.0013  | 28.7060745                           | 12.0527618   | 54.8613958  | 8.8581632  |
| 0.0014  | 31.4587393                           | 10.7139827   | 58.6405779  | 8.3308515  |

Table 8  
The benefit of fully integrated approach ( $\beta = 0.0012$ )

| $\theta$ | Proportion of the location costs (%) | The percentage improved by the fully integrated approach compared to the sequential approach (%) | Proportion of the location, inventory and order costs (%) | The percentage improved by the fully integrated approach compared to the partially integrated approach (%) |
|----------|--------------------------------------|--|---|--|
| 0.3      | 28.9237067                           | 11.2250058   | 55.4199814  | 7.3723689  |
| 0.4      | 27.8652472                           | 12.8872676   | 54.1591473  | 9.8267987  |
| 0.5      | 25.0890375                           | 14.4074944   | 52.606338   | 12.55916   |
| 0.6      | 22.0446327                           | 15.8831946   | 50.4708635  | 14.3697662   |
| 0.7      | 19.3196515                           | 17.4128125   | 47.6783391  | 16.3330685   |
| 0.8      | 17.4068934                           | 18.9134996   | 45.7672601  | 18.8493895   |

Table 9  
The benefit of fully integrated approach using genetic algorithm ( $\theta = 0.4$ )

| $\beta$ | Proportion of the location costs (%) | The percentage improved by the fully integrated approach compared to the sequential approach (%) | Proportion of the location, inventory and order costs (%) | The percentage improved by the fully integrated approach compared to the partially integrated approach (%) |
|---------|--------------------------------------|--|---|--|
| 0.0009  | 14.994003                            | 27.0422188   | 36.9493796  | 21.6172274   |
| 0.001   | 17.9567423                           | 15.2985484   | 40.7016556  | 12.5293527   |
| 0.0011  | 23.5504679                           | 10.6824252   | 48.5893693  | 7.8524233  |
| 0.0012  | 27.7926231                           | 8.6530055  | 54.0179946  | 5.9099714  |
| 0.0013  | 28.848198                            | 8.363913   | 55.1330141  | 5.8336611  |
| 0.0014  | 31.7191221                           | 7.6289569  | 59.1259436  | 4.7510939  |

In the above computational experiments and comparisons, we calculate the total routing distance using (7). Similar observations can also be obtained if we apply other methods to calculate the routing distance. For instance, Table 9 shows the benefit of integration when a genetic algorithm is applied to calculate the routing distance instead of (7). It is easy to see that the data in Table 9 reveal the same managerial insights as the data in Table 7 do.

## 6. Conclusion

In this paper, we have outlined a model for the stochastic supply chain design problem. To the best of our knowledge, this is the first paper that takes into consideration nonlinear inventory costs and routing costs when locating supply chain facilities. The model determines how many and where to locate DCs and how to assign customers to the DCs to minimize the total system costs, which include DC location costs, expected inventory costs at the DCs, and expected vehicle routing costs. This problem can be formulated as a nonlinear integer programming problem. By exploiting the structure of this problem, we are able to solve it efficiently for large sized problems.

The algorithm we provided can be applied to a wide range of other concave cost minimization problems. Thus, this algorithm can find application in other fields as long as the objective of the problem are separably concave.

The model proposed in this paper can be generalized to handle more general cases with distance or capacity constraints. It is also interesting to extend the model to a multiple-product setting.

## Acknowledgements

The authors are indebted to Editor Jacques Teghem and the referees for their helpful and valuable comments. This work was supported in part by NSF award DMI-0223323 and CAREER award DMI-0348209. This support is gratefully acknowledged.

## References

- Adelman, D., 2003. A price-directed approach to stochastic inventory/routing. Working Paper, University of Chicago.
- Agarwal, R., Ahuja, R.K., Laporte, G., Shen, Z.J., 2004. A composite very large-scale neighborhood search algorithm for the vehicle routing problem. In: Leung, J.Y-T. (Ed.), *Handbook of Scheduling: Algorithms, Models, and Performance Analysis*. ChapmanHall/CRC.
- Alonso-ayuso, A., Escudero, L.F., Garin, A., Ortuno, M.T., Perez, G., 2003. An approach for strategic supply chain planning under uncertainty based on stochastic 0–1 programming. *Journal of Global Optimization* 26, 97–124.
- Balakrishnan, A., Ward, J.E., Wong, R.T., 1987. Integrated facility location and vehicle routing models: Recent work and future prospects. *American Journal of Mathematical and Management Sciences* 7, 35–61.
- Camerini, P.M., Fratta, L., Maffioli, F., 1975. On improving relaxation methods by modified gradient techniques. *Mathematical Programming Study* 3, 26–34.
- Chakravarty, A.K., Orlin, J.B., Rothblum, U.G., 1985. Consecutive optimizers for a partitioning problem with applications to optimal inventory groupings for joint replenishment. *Operations Research* 33, 820–834.
- Christofides, N., Mingozzi, A., Toth, P., 1979. The vehicle routing problem. In: Christofides, N., Mingozzi, A., Toth, P., Sandi, C. (Eds.), *Combinatorial Optimization*. Wiley, Chichester.
- Daganzo, C., 1996. In: Beckmann, M., Krelle, W. (Eds.), *Logistics Systems Analysis, Lecture Notes in Economics and Mathematical Systems*. Springer Verlag, Berlin.
- Dasci, A., Verter, V., 2001a. A continuous model for production–distribution system design. *European Journal of Operational Research* 129, 287–298.
- Dasci, A., Verter, V., 2001b. The plant location and technology acquisition problem. *IIE Transactions* 33b, 963C974.
- Daskin, M., Coullard, C., Shen, Z.J., 2002. An inventory-location model: Formulation, solution algorithm and computational results. *Annals of Operations Research* 110, 83–106.
- Erlebacher, S.J., Meller, R.D., 2000. The interaction of location and inventory in designing distribution systems. *IIE Transactions* 32, 155–166.
- Goetschalckx, M., Vidal, C.J., Dogan, K., 2002. Modeling and design of global logistics systems: A review of integrated strategic and tactical models and design algorithms. *European Journal of Operational Research* 143, 1C18.
- Haimovich, M., Rinnooy Kan, A.H.G., 1985. Bounds and heuristics for capacitated routing problems. *Mathematics of Operations Research* 10, 527–542.

- Kleywegt, A., Nori, V.S., Savelsbergh, M.W.P., 2002a. The stochastic inventory routing problem with direct deliveries. *Transportation Science* 36, 94–118.
- Kleywegt, A., Nori, V.S., Savelsbergh, M.W.P., 2002b. Dynamic programming approximations for a stochastic inventory routing problem. Working paper, Georgia Institute of Technology.
- Langevin, A., Mbaraga, P., Campbell, J.F., 1996. Continuous approximation models in freight distribution: An overview. *Transportation Research* 30B, 163188.
- Min, H., Jayaraman, V., Srivastava, R., 1998. Combined location-routing problems: A synthesis and future research directions. *European Journal of Operational Research* 108, 1–15.
- Ozsen, L., 2004. Location-inventory planning models: Capacity issues and solution algorithms. Dissertation for the Degree of Doctor of Philosophy, Northwestern University.
- Santoso, T., Ahmed, S., Goetschalckx, M., Shapiro, A., 2005. A stochastic programming approach for supply chain network design under uncertainty. *European Journal of Operational Research* 167, 96–115.
- Shen, Z.J., 2000. Efficient algorithms for various supply chain problems. Ph.D. Dissertation, Northwestern University, Evanston, IL.
- Shen, Z.J., Coullard, C., Daskin, M., 2003. A joint location-inventory model. *Transportation Science* 37, 40–55.
- Shu, J., Teo, C.P., Shen, Z.J., 2005. Stochastic transportation-inventory network design problem. *Operations Research* 53, 48–60.
- Simchi-Levi, D., 1992. Hierarchical design for probabilistic distribution systems in euclidean spaces. *Management Science* 38, 198–211.
- Verter, V., Dincer, M.C., 1995. Facility location and capacity acquisition: An integrated approach. *Naval Research Logistics* 42, 1141C1160.
- Vidal, C.J., Goetschalckx, M., 2001. A global supply chain model with transfer pricing and transportation cost allocation. *European Journal of Operational Research* 129, 134–158.