African Journal of Business Management Vol. 6 (7), pp. 2678-2696, 22 February, 2012
Available online at http://www.academicjournals.org/AJBM
DOI: 10.5897/AJBM11.2201
ISSN 1993-8233 ©2012 Academic Journals

Full Length Research Paper

An integrated model for designing supply chain network under demand and supply uncertainty

Mir-Bahador Aryanezhad, Seyed Gholamreza Jalali Naini and Armin Jabbarzadeh*

Industrial Engineering Department, Iran University of Science and Technology, 16846113114 Tehran, Iran.

Accepted 2 June, 2011

This paper presents an integrated network design model for a supply chain in which supplier and distribution centers are unreliable. When there is an unreliable supplier, the amount of yield which is received at each distribution center may be different from the original orders. Similarly, because of imperfect performances of distribution centers, the quantity of products which is transformed from each DC to each customer may be less than what was originally planned. In such a system, customers have random demands and there is flexibility in determining which customers must be served. The proposed model of this study is formulated as a nonlinear integer programming to minimize the expected total cost which includes the costs of location, inventory, transportation and lost sales. The model simultaneously determines the optimal location of distribution centers, the subset of customers to serve, the assignment of customers to distribution centers and the cycle order quantities at distribution centers. In order to solve the resulted mathematical model, an efficient solution methodology based on Lagrangian relaxation approach and genetic algorithm is developed. Finally, computational results for several instances of the problem are presented to imply the effectiveness of the proposed approach.

Key words: Lagrangian relaxation, integrated supply chain design model, uncertainty, location, inventory, genetic algorithm, birth-death process.

INTRODUCTION

The efficient and effective design of supply chain is crucial in today's competitive environment. There is a growing realization that supply chain network design model is needed to determine the strategic decisions of location and the tactical decisions of inventory and transportation, simultaneously. Such integrated models in the literature typically assume that facilities and suppliers are always available and perform perfectly. However, in the real world cases, suppliers and facilities are vulnerable to randomly changing environmental conditions which may affect their performances.That is, there are varieties of sources like machine breakdown, raw material shortage, quality rejection, mistakes made during the assembly, workforce slow down, strike, requirements, parts shortage, loading or transportation maintenance duration, poor communication of customer damage and natural disaster leading to unreliable and uncertain performances of suppliers and facilities (Erdem and Ozekici, 2002; Wu, 2008).When suppliers and facilities do not perform perfectly, a supply chain may lose its customers and it may be faced with huge amount of lost sales costs. In other words, the imperfect performances of suppliers and facilities can be costly and can bring a supply chain to a screeching halt. For instance, Boeing experienced supplier delivery failure of two components, with an estimated loss to the company of $2.6 billion (Radjou, 2002).Similarly, Hurricane Katrina and Rita resulted in shutdowns of numerous facilities and consequent significant economic losses (Barrionuevo and Deutsch, 2005).These examples highlight the need for supply chain design models that account for imperfect performances of suppliers and facilities. Such models are required to design the supply chain network in a way that the costs of imperfect performances of suppliers and facilities are reduced. The majority of the integrated supply chain design models in the literature are based on

*Corresponding author. E-mail: arminj@iust.ac.ir. Tel: (+98 912) 336 99 24. Fax: (+98 21) 2277 33 61.

Abbreviation: DC, Distribution center.

the unrealistic assumption that the demand for all customers must be provided. However, profit-maximizing companies in practice prefer to lose their potential customers when the costs of maintaining the customers are prohibitive (Shen, 2006).Likewise, in many cases supply chains are inevitably unable to satisfy all the demands due to unreliable performances of their suppliers and facilities. Therefore, it is of particular importance to develop supply chain design models which consider the possibility of not serving all the customers and losing some customers' demands. This paper presents an integrated design model for a supply chain which consists of a supplier, distribution centers (DCs) and customers. The supplier and DCs are assumed to be unreliable. The proposed model of this study considers the uncertainties arising from random demands of customers and unreliable performances of the supplier and DCs. Specifically, the supplier ships one type of product to customers in order to provide their uncertain demands. DCs function as the direct intermediary between the supplier and customers for the shipment of the product. Namely, DCs combine the orders from different customers and then order to the supplier. Due to unreliable performance of the supplier, the amount of yield which is received at each DC may be different from what was ordered. Similarly, because of imperfect performance of DCs, the quantity of products which is transformed from each DC to each customer may be less than what was originally planned. When the amount of yield at each customer is less than the due quantity, the system incurs penalty costs. Another key characteristic of the problem is the flexibility of supply chain in deciding which customers to serve. In fact, there is no restrictive assumption that all the customers' demands have to be met. Thus, when the cost of serving any customer is prohibitively high, the supply chain may choose to incur lost sales costs and not to serve that customer at all. The problem lies in simultaneously determining: 1) where DCs are located; 2) which customers are served; 3) which DCs are assigned to which customers; 4) how much and how often to order at each DC. We formulate the problem as a nonlinear integer programming model which minimizes expected total cost. The total cost includes fixed location costs, inventory costs at the DCs, shipment costs, penalty costs of unreliable performances of DCs and lost sales costs An efficient solution method incorporating Lagrangian relaxation approach and genetic algorithm is adopted to solve the proposed nonlinear integer programming model. This study reviews the related models in the literature briefly. In fact, the literature on integrated supply chain design models as well as the literature on facility location planning under the risk of disruptions is reviewed. Afterwards, the assumptions underlying the problem are explained and the necessary parameters for formulating the model are stated. Next, each cost component of the problem is formulated and the integrated model is presented, after which a solution method based on Lagrangian relaxation is developed to solve the proposed model. Here, the study shows how to obtain upper and lower bounds for the integrated model. The results of computational experiments with the proposed solution approach are then presented. Three sets of experiments are designed. The first experiment evaluates the performance of the proposed solution approach using popular data sets in the literature. The second experiment compares the performance of the presented solution approach with simulated annealing. The focus of the third experiment is on the benefits of considering the supply uncertainty during the supply chain design phase. Finally, the study is concluded along with directions for future research.

LITERATURE REVIEW

It begins with a brief literature review of the integrated supply chain design models which jointly determine location and inventory decisions. For reviews on the recent models addressing inventory decisions, readers can refer to Baten and Kamil (2009), chen et al. (2010), Wazed et al. (2010a, b) and Cheng and Ting (2010). A comprehensive literature survey has been performed by Shen (2007) and Melo et al. (2009) to study location decisions in the context of supply chain management. Being aware that ignoring interaction between long and short terms decisions can result in sub-optimality (Shen and Qi, 2007; Shu et al., 2005; Ozsen, 2004), researchers have concentrated on the integrated supply chain design models with nonlinear terms. Erlebacher and Meller (2000) present an integrated location-inventory model to design a two-level distribution system serving continuously represented customer locations. They use heuristic procedures to solve their integrated model. Shen (2000), Shen et al. (2003) and Daskin et al. (2002) develop a location model with risk pooling (LMRP) which includes inventory and location decisions in the same model. The objective of LMRP is to minimize the sum of facility location costs, linear shipment and nonlinear inventory costs. In order to solve LMRP, Shen (2000) and Shen et al. (2003) apply column generation, while Daskin et al. (2002) use Lagrangian relaxation. Another efficient approach to solve the LMRP is proposed by Shu et al. (2005). LMRP is extended by Shen and Daskin (2005), Ozsen et al. (2008), Shen and Qi (2007), and Snyder et al. (2007). Shen and Daskin (2005) investigate an integrated location-inventory model with customer service consideration and develop practical approaches for evaluation of cost/service trade-offs. Shen and Qi (2007) add routing decisions to the LMRP framework; that is, they examine an integrated model which determines location, inventory and routing decisions, simultaneously. The stochastic version of LMRP is introduced by Snyder et al. (2007). Their model handles uncertainty by defining discrete scenarios and minimizes the expected system cost across all scenarios. Other related supply chain design models are studied by Shen (2006), Sourirajan et al. (2007, 2008) and Ozsen et al. (2009). Shen (2006) presents a profit-maximizing supply chain network design model where each DC can charge different prices to explore the willingness to buy in different regions. The author assumes that every customer has a reserve price and the company loses a customer if the total price is higher than the customer's reserve price. Sourirajan et al. (2007) investigate the two-stage supply chain with a production facility in which the replenishment lead time at a DC depends on the volume of flow through the DC. They model the relationship between the flows in the network, lead times,and safety stock levels and use a Lagrangian heuristic to obtain near-optimal solutions for the proposed model. Sourirajan et al. (2008) propose genetic algorithm to solve the model and imply that the genetic algorithm outperforms the Lagrangian heuristic developed in the earlier work. Ozsen et al. (2009) analyze the impact of multi-sourcing by introducing a location-inventory model that minimizes the sum of the fixed location costs, the transportation costs, and the inventory costs. All of the aforementioned integrated models assume that suppliers and DCs perform perfectly. Another relevant body of literature is the literature on facility location with disruptions. Snyder and Daskin (2005) address facility location problem when facilities fail with a given fixed probability. They present models for choosing facility locations to minimize the weighted sum of two objectives. The first objective indicates the cost of the system when no disruptions occur, whereas the second objective represents the expected transportation cost after accounting for disruptions. Their models rely on the assumption that the failure probabilities of the facilities are equal. This assumption is released by Berman et al. (2007). In fact, they develop a similar P-median model, in which the facilities can have different probabilities of failures. Since their nonlinear model is not tractable, they suggest using heuristic methods to solve the problem. Similarly, Church and Scaparra (2007) and Scaparra and Church (2008) study models for facility location with disruptions. They focus on facility location problems where existed facilities can be protected against disruptions by limited fortification resources. They develop models determining what facilities are protected, in order to minimize the impact of interdiction on the remaining system operation. Snyder and Daskin (2007) examine facility location models under a variety of risk measurements and operating strategies. Snyder et al. (2006) provide a tutorial which reviews a broad range of models for facility location with disruptions. Among the aforementioned works in the context of facility location with disruptions, no model considers inventory costs. Other related models are provided by Qi and Shen (2007), Lim et al. (2009), Cui et al. (2010), Aryanezhad et al. (2010) and Qi et al. (2010). Qi and Shen (2007) and Aryanezhad et al. (2010) study joint location-inventory models with unreliable facilities. However, they assume that the supplier is perfectly reliable and facilities will always receive the exact amount they order. Qi et al. (2010) propose an integrated location-inventory model in which the supplier and retailers are disrupted randomly. Their model assumes that the demands are deterministic and the lead time for order processing is zero. They use an effective approximation of the objective function, in order to analyze and to solve the model through common solution algorithm. The present paper is different from the earlier works in the literature of supply chain design network in some main directions. First, unlike the most of supply chain design models in the literature, this study considers the uncertainties of customers' demands, yields of the supplier and DCs' performances, simultaneously. In other words, this article considers demand and supply uncertainty in the same model. Furthermore, the model proposed in this work dismisses the common unrealistic assumption in the literature that all the customers' demands have to be satisfied. In fact, the model considers lost sales costs along with location, inventory and transportation costs. Finally, this research develops an effective solution method by incorporating genetic algorithm and Lagrangian relaxation approach.

METHODS

Model formulation

This formulates an integrated model for the problem stated in the introduction. The proposed model simultaneously determines which customers are served, where DCs are located, which DCs are assigned to which customers, and how much products each DC orders to the supplier. The objective is to minimize the expected total cost including: 1) the fixed costs of locating DCs, 2) the inventory costs at DCs, 3) transportation costs from DCs to customers, 4) the lost sales costs of not selecting some customers to serve at all, 5) the penalty costs for unreliable performances of DCs. In the following, first, the assumptions underlying the model are explained. In addition, the notations used for formulating the model are stated. Then, each cost component of the problem is formulated and the integrated model is proposed.

Assumptions

The model is based on the following assumptions:

1) The customers' demands are independent and follow a Poisson process. This is a common assumption in the literature (Daskin et al., 2002; Shen et al., 2003; Ozsen, 2004; Ozsen et al., 2008).
2) The supply chain is flexible in determining which customers to serve. For this case a customer is not served at all, lost sales cost is incurred. This assumption makes the model more applicable for companies in the competitive environments (Shen, 2006).
3) In order to make the model more practical, it is assumed that the supplier is not always reliable. Being unreliable simply means that the supplier may be unable to provide the order of a DC perfectly. As a result, the amount of provided products for a DC may be less than what the DC originally orders. In other words, the supplier has two different modes for each DC: reliable mode and unreliable

mode. At the reliable mode the supplier is able to perfectly provide the order placed by a DC. However, at the unreliable mode the supplier can provide only a fraction of the order placed by the DC.
4) In order to make the model more realistic, it is assumed that DCs are not always reliable. In other words, each DC has two states: reliable state and unreliable state. When a DC is at the reliable state it can satisfy all the demands of its customers. However, when the DC is at the unreliable state, it can satisfy only a fraction of the customers' demands. In this case, penalty cost is incurred for the unsatisfied fraction of the customers' demands.
5) The durations of the reliable and unreliable modes of the supplier are uncertain and follow independent exponential distributions. Likewise, the durations of the reliable and unreliable states of the DCs are assumed to follow independent exponential distributions. It is believed that the exponential distribution is reasonable in this context, since exponential distributions are often applied to model the time between independent events that happen at a constant average rate, and otherwise are often suitable approximations to the actual distributions (Ross, 2007; Qi et al., 2009).

Notations

To develop the model the following parameters and decision variables are used. Additional notations will be given out when needed.

Parameters

I : Set of potential customers indexed by *i*,
J : Set of candidate locations for distribution centers indexed by *j*,
f_j : Fixed cost of locating a DC at *j*, for each $j \in J$,
F_j : Fixed cost of placing an order at DC at *j*, for each $j \in J$,
g_j : Fixed cost per shipment from the supplier to DC at *j*, for each $j \in J$,
A_j : Per-unit shipment cost from the supplier to DC at *j*, for each $j \in J$,
q_j : Fraction of the order placed by DC at *j* which can be provided by the supplier when the supplier is at the unreliable mode in regard to DC at *j*, for each $j \in J$,
a_j : Duration rate of the supplier's unreliable mode in regard to DC at *j*, for each $j \in J$,
b_j : Duration rate of the supplier's reliable mode in regard to DC at *j*, for each $j \in J$,
r_j : Fraction of the assigned demands to DC at *j* which can be satisfied when it is at the unreliable state, for each $j \in J$,
h : Inventory holding cost per unit of product,
D_i : Mean of demand at customer *i*, for each $i \in I$,
e : Penalty cost for losing a unit of demand of customers due to the unreliable performances of DCs,
w_j : Exponential rate which DC at *j* leaves the reliable state for each $j \in J$,
v_j : Exponential rate which DC at *j* leaves the unreliable state for each $j \in J$,
d_{ij} : Per-unit cost to ship from DC at *j* to customer *i*, for each $i \in I$ and for each $j \in J$,*P*: Number of DCs which must be located, s_i : Lost sales cost of deciding not to serve customer *i* at all, per unit of demand for each $i \in I$, β: Weight factor associated with the shipment cost, θ: Weight factor associated with the inventory cost.

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 assigned to a DC based at *j*, and 0 otherwise, for each $i \in I$ and $j \in J$.

Inventory cost

This formulates the expected inventory cost at each located distribution center *j*, for each $j \in J$. As stated in the assumptions, the supplier has two different modes for each DC: reliable mode and unreliable mode. At the reliable mode the supplier is able to provide all the order placed by a DC, whereas at the unreliable mode it can provide only a fraction of the order placed by the DC. Specifically, when the supplier is at the unreliable mode for DC at *j*, it can provide $q_j\%$ of the order placed by this DC. The durations of the unreliable and reliable modes of the supplier for DC at *j* follow the independent exponential distributions with rates a_j and b_j , respectively. Let Q_j be the unknown reorder quantity of distribution center *j* and $[Q_j q_j]$ represents integer value of $Q_j \times q_j\%$ .Then, when the supplier is at the reliable mode the amount of yield at distribution center *j* will be Q_j .However, when the supplier is at the unreliable mode the amount of yield at distribution center *j* will be $[Q_j q_j]$ .Due to finite number of inventory at distribution center *j* and memory-less property of exponential distribution, we can define inventory quantities as states, and the inventory transition can be modeled as a birth-death process demonstrated in Figure 1 (Wu, 2008). Note that μ_j indicates the unknown demand arrival rate to distribution center *j* in Poisson process(in the subsequent study, we show how μ_j can be obtained based on the decision variables and parameter D_i). In order to gain the expected inventory cost, we need to gain the limiting probabilities of states of the birth-death process.The limiting probabilities of the states can be obtained by equating the rate at which the process leaves a state with the rate at which it enters that state as follows (Ross, 2007; Wu, 2008):

$$\pi(1) = \pi(2) = \ldots = \pi([Q_j q_j]) = \frac{1}{p_1[Q_j q_j] + p_2 Q_j} \tag{1}$$

$$\pi([Q q_j] + 1) = \pi([Q_j q_j] + 2) = \ldots = \pi(Q_j) = \frac{p_2}{p_1[Q_j q_j] + p_2 Q_j} \tag{2}$$

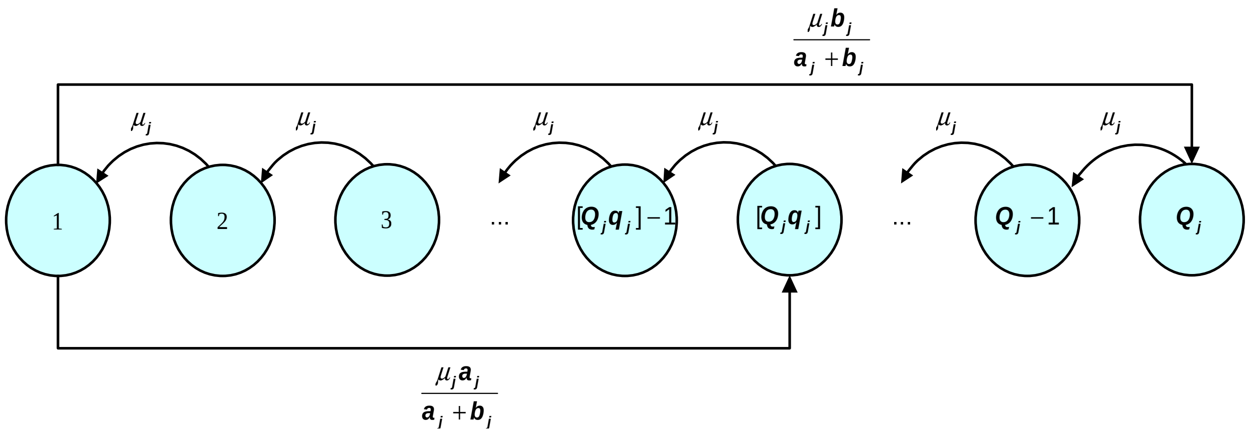

Figure 1. A birth-death process for inventory transition of distribution center at *j*.

Where $\pi(k)$ indicates the limiting probability of state *k*, for *k* = 1 to Q_j (Figure 1). Also, $p_1 = \frac{a_j}{a_j + b_j}$ and $p_2 = 1 - p_1$. Therefore, the expected inventory cost at distribution center *j* can be obtained by:

$$F_j N_j + \beta(g_j N_j + A_j \mu_j) + \theta h \sum_{k=1}^{Q_j} k\,\pi(k) \qquad (3)$$

Where N_j indicates the number of orders and is equal to

$$\frac{(a_j + b_j)\mu_j}{a_j [Q_j q_j] + b_j Q_j}$$ (Wu, 2008).

The first term of Equation 3 represents the fixed cost of placing orders. The second term is the cost of shipping orders from the supplier to the DC at *j*, assuming that the shipment cost from the supplier to distribution center *j* has a fixed cost g_j and volume dependent cost A_j. The last term indicates the cost of holding average of $\sum_{k=1}^{Q_j} k\,\pi(k)$ units of inventory. Substituting limiting probabilities in Equations 1 and 2 into Equation 3, the inventory cost at the distribution center *j* is obtained as follows:

$$\frac{100(F_j + \beta g_j)\mu_j (a_j + b_j)}{q_j a_j Q_j + b_j Q_j} + \frac{\theta h q_j (q_j Q_j + 100) + h p_2 (Q_j + q_j Q_j + 100)(1 - q_j)}{200(q_j p_1 + p_2)} + \beta A_j \mu_j \qquad (4)$$

In order to determine the optimal reorder quantity, we take derivative of Equation 4 in respect to Q_j and set the derivative to zero. By this way the optimal value will be gained by:

$$Q_j = 100\sqrt{\frac{2(F_j + \beta g_j)(a_j + b_j)}{\theta h (q_j^2 a_j + b_j)}} \qquad (5)$$

Plugging Equation 5 into Equation 4, inventory cost at the distribution center *j* will be obtained as follows:

$$\frac{1}{q_j a_j + b_j}\sqrt{2(F_j + \beta g_j)\mu_j \theta h (q_j^2 a_j + b_j)(a_j + b_j)} + \beta A_j \mu_j \qquad (6)$$

Penalty cost for unreliable performance of a distribution center

As stated earlier, each DC has two states as follows: reliable state and unreliable state. When the DC at *j* is at the reliable state it can satisfy all the demands of its customers. However, when the DC is at the unreliable mode, it can provide only $r_j\%$ of the customers' demands and $(1 - r_j)\%$ of the demands are unmet. It is assumed that distribution center *j* leaves the reliable state and unreliable state exponentially with the rates w_j and v_j, respectively, as shown in Figure 2. Considering the memory-less property of the exponential distribution, we equate the rate at which the process leaves a state with the rate at which it enters that state and obtain the limiting probabilities of the states:

$$\pi(R) = \frac{v_j}{w_j + v_j} \qquad (7)$$

$$\pi(UR) = \frac{w_j}{w_j + v_j} \qquad (8)$$

Where $\pi(R)$ and $\pi(UR)$ denote the limiting probabilities of the

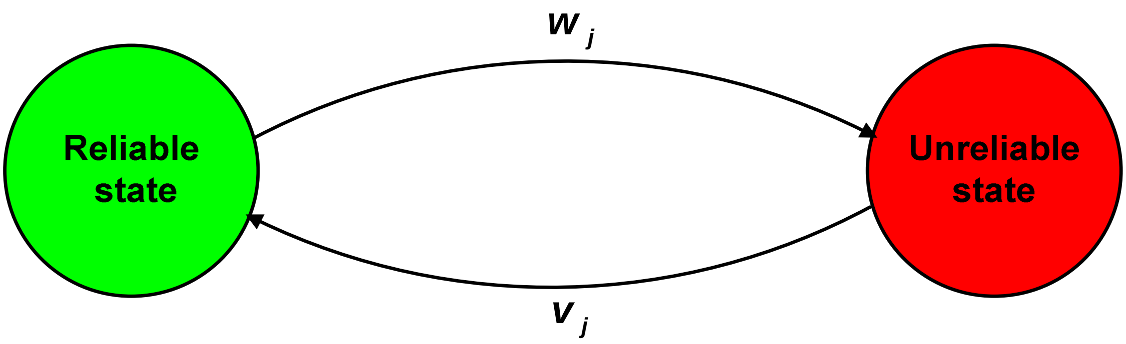

Figure 2. A state transition diagram for distribution center at *j*.

reliable state and unreliable state, respectively. Let e be the penalty cost for losing a unit of demand due to the unreliable performance of the DC at *j*. In addition μ_j denotes the total demand allocated to the DC at *j*. Then, the expected penalty cost for unreliable performance of the DC at *j* will be:

$$\pi(UR)\times e\times\mu_j\times(1-r_j)\%+\pi(R)\times e\times\mu_j\times 0\%$$
$$=\frac{w_j}{w_j+v_j}\times e\times\mu_j\times(1-r_j)\% \tag{9}$$

Integrated model

In order to formulate the integrated model, two sets of decision variables are used:

i) $X_j=1$, if *j* is selected as a DC location, and 0, otherwise, for each $j\in J$,

ii) $Y_{ij}=1$, if customer *i* is assigned to a DC based at *j*, and 0 otherwise, for each $i\in I$ and $j\in J$.

At this stage, the total demand assigned to the distribution center at *j* and μ_j can be written in terms of decision variables:

$$\mu_j=\sum_{i\in I}D_iY_{ij}$$

To model the lost sale cost of deciding not to serve customers, it is expedient to define a dummy DC with index *z*. Assigning the customer *i* to this dummy, DC ($Y_{iz}=1$) indicates not serving customer *i* at all (Snyder and Daskin, 2005).

Regarding dummy distribution center *z*, we assume that it has the shipment cost $d_{iz}=s_i$ to customer $i\in I$ and there is no other cost. Therefore, when customer $i\in I$ is assigned to dummy distribution center *z* it means that the customer *i* is not served at all and the lost sale cost is incurred. From this point forward, the dummy distribution center *z* is added to set J. In addition, it is forced that $X_z=1$. With this notation, the problem is formulated as follows:

$$\begin{aligned}&\mathrm{Min}\sum_{j\in J}f_jX_j+\left(\beta\sum_{j\in J}\sum_{i\in I}d_{ij}D_iY_{ij}\right)\\&+\left(\sum_{j\in J}\frac{1}{q_ja_j+b_j}\sqrt{2(F_j+\beta g_j)\theta h(q_j^2a_j+b_j)(a_j+b_j)\sum_{i\in I}D_iY_{ij}}\right.\\&\left.+\beta\sum_{j\in J}A_j\sum_{i\in I}D_iY_{ij}\right)\\&+\sum_{j\in J}\frac{w_j}{w_j+v_j}\times(1-r_j)\sum_{i\in I}eD_iY_{ij}\end{aligned} \tag{10}$$

Subject to:

$$\sum_{j\in J}Y_{ij}=1\quad\forall i\in I \tag{11}$$

$$Y_{ij}\le X_j\quad\forall i\in I,\ \forall j\in J \tag{12}$$

$$X_z=1 \tag{13}$$

$$\sum_{j\in J}X_j=P+1 \tag{14}$$

$$X_j\in\{0,1\}\quad\forall j\in J \tag{15}$$

$$Y_{ij}\in\{0,1\}\quad\forall i\in I,\ \forall j\in J \tag{16}$$

The objective function of Equation 10 is composed of four components. The first component indicates the fixed cost of locating DCs. The second part represents the expected shipment cost from the DCs to customers. Recall that set J includes dummy distribution center *z*, in order to take lost sales costs into account in the model. Furthermore, note that $\sum_{i\in I}D_iY_{ij}$ indicates the total demand allocated to the distribution center at *j*. Therefore, the third component represents the inventory cost of Equation 6 where $\mu_j=\sum_{i\in I}D_iY_{ij}$. Considering Equation 9, the forth part indicates the expected penalty costs for unreliable performances of the DCs. Constraints 11 require that each customer is assigned to a

DC. Recall that allocating a customer to dummy distribution center *z* is equivalent to choose not to serve the customer at all and to incur lost sales costs. Constraints 12 state that customers can only be allocated to candidate sites that are selected as DCs. Constraint 13 stipulates that the dummy distribution center *z* is located. Constraint 14 requires that the number of located at DCs is exactly *P* + 1 (this means that *P* distribution centers must be located in addition to dummy distribution center *z*). Constraints 15 and 16 are binary constraints. Objective function of Equation 10 can be reorganized as follows:

$$\begin{aligned}
&\sum_{j\in J} f_j X_j + \left(\beta \sum_{j\in J}\sum_{i\in I} d_{ij} D_i Y_{ij}\right) + \\
&\left(\begin{aligned}&\sum_{j\in J}\frac{1}{q_j a_j + b_j}\sqrt{2(F_j+\beta g_j)\theta h(q_j{}^2 a_j + b_j)(a_j+b_j)\sum_{i\in I} D_i Y_{ij}} \\ &+\beta\sum_{j\in J} A_j \sum_{i\in I} D_i Y_{ij}\end{aligned}\right) \\
&\quad + \sum_{j\in J}\frac{w_j}{w_j+v_j}\times(1-r_j)\sum_{i\in I} e D_i Y_{ij} \\
&\quad = \sum_{j\in J}\left\{f_j X_j + \sum_{i\in I}\tilde{d}_{ij} Y_{ij} + k_j\sqrt{\sum_{i\in I} D_i Y_{ij}}\right\}
\end{aligned} \tag{17}$$

Where:

$$\tilde{d}_{ij} = \left[\beta(d_{ij} + A_j) + \frac{w_j}{w_j+v_j}\times(1-r_j)\times e\right] D_i$$

$$k_j = \frac{1}{q_j a_j + b_j}\sqrt{2(F_j+\beta g_j)\theta h(q_j{}^2 a_j + b_j)(a_j+b_j)}$$

Solution method

In order to solve the proposed model, a Lagrangian relaxation approach (Fisher, 1981, 1985) is used. Lagrangian relaxation approach is capable of providing both upper and lower bounds on the optimal value of the objective function. That is, this method allows the decision maker to know how far from the optimality the best found feasible solution is (Current et al., 2001). The detailed solution approach including finding lower bound and upper bound for the model is explained here.

Finding a lower bound

Relaxing constraints of Equation 11 with Lagrange multipliers, λ_i, obtains the following Lagrangian dual problem:

$$\begin{aligned}
&\underset{\lambda}{\text{Max}}\ \underset{X,Y}{\text{Min}} \sum_{j\in J}\left\{f_j X_j + \sum_{i\in I}\tilde{d}_{ij} Y_{ij} + k_j\sqrt{\sum_{i\in I} D_i Y_{ij}}\right\} \\
&\qquad + \sum_{i\in I}\lambda_i(1-\sum_{j\in J} Y_{ij}) \\
&= \sum_{j\in J}\left\{f_j X_j + \sum_{i\in I}(\tilde{d}_{ij}-\lambda_i) Y_{ij} + k_j\sqrt{\sum_{i\in I} D_i Y_{ij}}\right\} \\
&\quad + \sum_{i\in I}\lambda_i
\end{aligned} \tag{18}$$

Subject to:

$$Y_{ij} \le X_j \ \ \forall i\in I,\ \forall j\in J \tag{19}$$

$$X_z = 1 \tag{20}$$

$$\sum_{j\in J} X_j = P + 1 \tag{21}$$

$$X_j \in \{0,1\} \ \ \forall j \in J \tag{22}$$

$$Y_{ij} \in \{0,1\}\ \forall i\in I,\ \forall j \in J \tag{23}$$

For given values of the Lagrange multipliers, λ_i, the objective is to minimize Equation 18 over the decision variables X_j and Y_{ij}. This problem can be decomposed by *j*; thus, we need to solve the following sub-problem for each candidate location $j \in J$:

$$\text{SP}_j : \tilde{V}_j = \text{Min} \sum_{i\in I} l_i Y_i + \sqrt{\sum_{i\in I} u_i Y_i} \tag{24}$$

Subject to:

$$Y_i \in \{0,1\} \ \ \forall i \in I \tag{25}$$

Where $l_i = \tilde{d}_{ij} - \lambda_i$ and $u_i = k_j^2 D_i$. In Equations 24 to 25, the assignment variables Y_{ij} have been replaced by Y_i to simplify the notation, as SP_j is specific to distribution center *j*. Sub-problem SP_j can be solved efficiently applying the exact algorithm introduced by Shen et al. (2003). Customized to our problem, their algorithm is as follows:

1) Define $I^0 = \{i : l_i < 0 \text{ and } u_i = 0\}$ and $I^- = \{i : l_i < 0 \text{ and } u_i > 0\}$.

2) Calculate the values of l_i / u_i for the elements of I^-.

3) Sort the elements of I^- in increasing order of l_i / u_i and indicate the elements by $1^-, 2^- \ldots n^-$, respectively, where $n = |I^-|$.

4) Find the value of m that minimizes:

$$\sum_{i\in I^0} l_i Y_i + \sqrt{\sum_{i\in I^0} u_i Y_i} + \sum_{i=1,\, i\in I^-}^{m} l_i Y_i + \sqrt{\sum_{i=1,\, i\in I^-}^{m} u_i Y_i}\ .$$

5) The optimal solution to sub-problem SP_j is gained by $Y_i = 1$ for $i \in I^0$, $Y_{1^-} = Y_{2^-} = \ldots = Y_{m^-} = 1$ for $i \in I^-$ and $Y_i = 0$ for all other $i \in I$. When SP_j for each $j \in J$ is solved, f_j is added to the optimal objective value of $\tilde{V}_j$. Then, $\tilde{V}_j$ values are sorted from the smallest to the largest for all $j \in J$, excluding

X_1	X_2	X_3	X_4	X_5	Y_1	Y_2	Y_3	Y_4
1	0	1	0	1	1	5	3	1

Figure 3. Chromosome structure.

dummy distribution center *z*. The first P values of V_j are identified and the corresponding X_j variables are set to 1. Also, we set $X_z = 1$. For each chosen distribution center *j* (those for which $X_j = 1$) the assignment variables Y_{ij} are the same as the optimal Y_i values in sub-problem SP_j. But, for each unselected distribution center *j* (those for which $X_j = 0$) $Y_{ij} = 0, \forall i \in I$ .Having solved the Lagrangian problem, the optimal Lagrange multipliers are obtained using a standard sub-gradient optimization procedure (Fisher, 1981, 1985). The optimal objective value of the Lagrangian dual problem Equation 18 can provide a lower bound on the optimal objective value of Equation 17.

Finding an upper bound

It is extremely hard to solve the presented nonlinear and stochastic model in a reasonable time. For instance, even solving the presented model in the simplest condition (when $f_j = 0,\ k_j = 0$ for each $j \in J$) is identical to solving the P-median problem which is NP-hard (Garey and Johnson, 1979). In this context, genetic algorithm (GA) can overcome computational complexity caused by the nonlinear and stochastic objective function to solve the model (Min et al., 2006; Sourirajan et al., 2008). GA has been successfully used in various facility location and supply chain network design problems and has proven to be a very effective heuristic procedure to solve these problems, particularly problems of large scale (Jaramillo et al., 2002; Alp et al., 2003; Drezner and Wesolowsky, 2003; Shen and Daskin, 2005; Snyder and Daskin, 2006; Sourirajan et al., 2008). For these reasons, an effective heuristic based on GA is developed in order to obtain a suitable upper bound at each iteration of the Lagrangian procedure. GA is a stochastic solution search method based on the mechanism of natural genetics, which starts with an initial set of potential solutions to the problem, called a population. Each individual solution in the population is known as chromosome and each component of the chromosome is named gene. The chromosomes evolve through successive iterations, called generation. The population of the next generation includes some chromosomes of the current population and some new chromosomes. To determine which chromosomes of the current population are selected for the population of the next generation, each chromosome in the current population is evaluated using some measure of fitness. Fitter chromosomes have higher probabilities of being selected for the next generation. In order to create the new chromosomes (called offspring) for the population of the next generation, crossover and mutation operators are used. Crossover operation selects two chromosomes from the current population at random and combines them to form offspring. Though, mutation process creates an offspring by altering the genes of a single chromosome. After several generations, the algorithm converges to the best chromosome, which can represent the optimum or near optimal solution to the problem (Gen and Cheng, 1996, 2000). For comprehensive review of GA and its application in location problem refer to Gen and Cheng 1996, Sourirajan et al. (2008), Goldberg (1989) and Jaramillo et al. (2002). The following study explains the developed GA for finding an upper bound.

Encoding

In the proposed GA, each chromosome is represented as a single dimensional array demonstrating decision variables. Let *n* be the number of candidate, DCs and *m* be the number of customers. Then, each chromosome *C* can be indicated by:

$$C = (X_j,\ Y_i) = (X_1,\ X_2 \ldots X_n,\ X_{n+1},\ Y_1,\ Y_2 \ldots Y_m).$$

Where X_j corresponds to the location genes and Y_i corresponds to the assignment genes. These genes represent where the DCs are located and how the customers are assigned to the located DCs, respectively. In other words, $X_j = 1$ means that candidate site *j* is chosen as a DC location, whereas $X_j = 0$ shows that candidate location *j* is not selected as a DC site.The gene X_{n+1} corresponds to dummy distribution center *z*; as a result, it always takes the value 1. Thus, the location genes demonstrate location decision variables. Also, $Y_i = j$ indicates that customer *i* is assigned to distribution center *j*. If customer *i* is assigned to the dummy distribution center *z*, the corresponding assignment gene takes the value of *n*+1; that is, $Y_i = n+1$. Therefore, the values of the assignment decision variables can be known by the assignment genes. For example, in Figure 3, distribution centers are located at 1 and 3. It follows that customers 1 and 4 are assigned to the DC at 1 and customer 3 is allocated to the DC at 3. Also, customer 2 is assigned to the DC at 5, which corresponds to the dummy distribution center *z*.

Generating the first population

The chromosomes of the first population are generated by two methods. In the first method, the chromosomes of the first population are generated from the feasible region randomly. The second method forms the chromosomes by modifying the lower bound solution in the Lagrangian procedure. That is, it identifies customers that are assigned to more than one DC in the lower bound solution of the Lagrangian procedure. Then, such customers are assigned to exactly one DC which is selected randomly. By this way, the obtained lower bound solution in the Lagrangian procedure can be modified to feasible solutions and the needed chromosomes for the first population are formed. The numbers of the chromosomes generated by the first and the second methods are

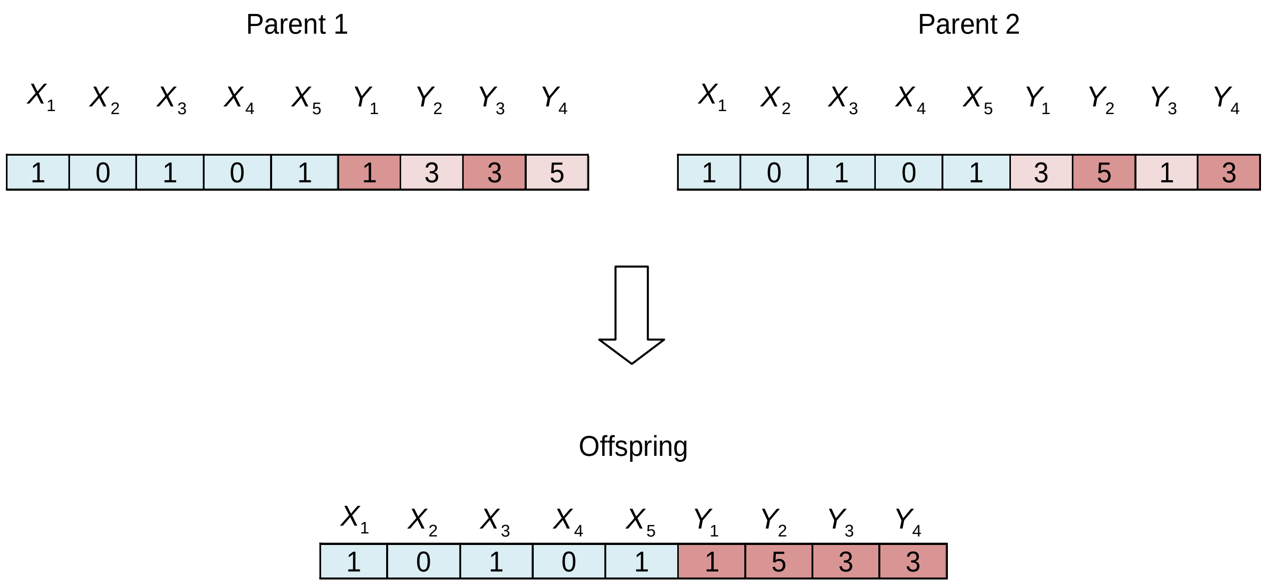

Figure 4. Sample of crossover.

the same.

Fitness function

The rank-based evaluation function is defined as the objective function of Equation 17 for the chromosomes. In fact, the value of the objective function of Equation 17 is calculated for each of the chromosomes. Obviously, the chromosomes which lead to less values of objective function of Equation 17 have the better rank.

Crossover process

Crossover operator generates offspring by combining two random chromosomes, called parents. Let C_k denote the chromosomes of the population for *k* = 1, 2 … pop-size. In order to determine which of these chromosomes are selected for crossover operation, the following practice is repeated from *k* = 1 to pop-size. A random number *r* from the interval (0, 1) is generated. Chromosome C_k will be chosen for crossover process provided that $r < P_C$, where the parameter P_C is the probability of crossover. Then selected parents C_1', C_2', C_3', … are grouped randomly to the pairs (C_1', C_2'), (C_3',C_4'),…. Without loss of generality let us outline the crossover operator on each pair by (C_1',C_2'). Crossover operator allocates each customer *i* in offspring chromosome either to the DC which is assigned to customer *i* in chromosome C_1', or to the DC which is assigned to customer *i* in chromosome C_2'. This occurs randomly and with probability of 0.5. The obtained offspring can be infeasible. If a customer is assigned to an unselected candidate DC location, this infeasibility is removed by locating DC in that candidate site. In case that the number of located DCs exceeds $P+1$, the number of located DCs is reduced to $P+1$ by closing some DCs randomly. The customers which are assigned to the closed DCs are allocated randomly to one of the located DCs. By this way, the offspring can be modified to a feasible chromosome. A sample of crossover operator is shown in Figure 4.

Mutation process

Mutation operator modifies a chromosome to form offspring. In order to decide which of chromosomes C_k undertake mutation, the following process is repeated for *k* = 1 to pop-size. A random number *r* from the interval (0, 1) is generated. Then, the chromosome C_k will undergo mutation process provided that $r < P_M$, where the parameter P_M is the probability of mutation. Chosen chromosomes are altered by one of the two following types of mutation for several times. In the first type of mutation, offspring is generated by modifying the assignment genes of the parent chromosome. In other words, the first type of mutation selects two located DCs randomly; let *s* and *t* denote them. Then, if any customer in parent chromosome is allocated to *s*, that customer will be allocated to *t* and if any customer is assigned to *t*, it will be assigned to *s*. The second type of mutation modifies location genes of the parent chromosome to create offspring. In fact, the second type of mutation randomly chooses a location in which no DC is located; let *t* denotes such location. Next, a DC is chosen randomly from the located DCs and is named *s*. This type of mutation closes distribution center *s* and instead of it locates a DC at candidate site *t*. Then, all the customers allocated to distribution center *s* are assigned to distribution center *t*. The samples of mutation type 1 and mutation type 2 are illustrated in Figures 5 and 6, respectively.

RESULTS AND DISCUSSION

Here, the results of computational experiments with the outlined Lagrangian relaxation approach. The solution approach was coded in Visual Basic.Net and executed on

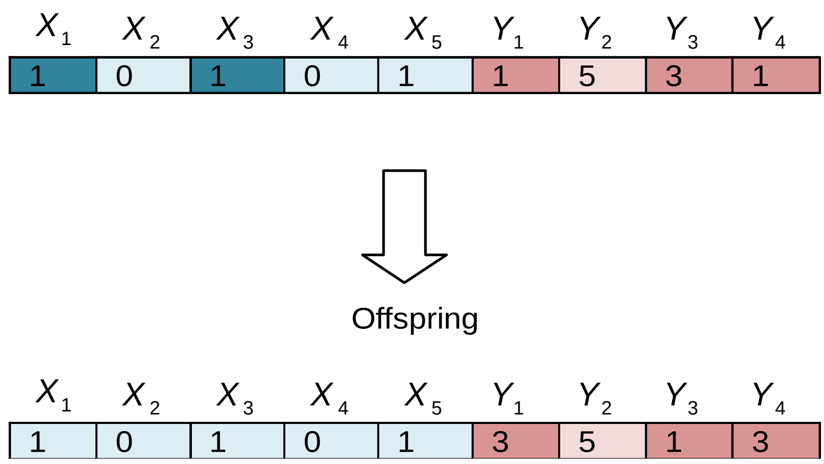

Figure 5. Sample of mutation type 1

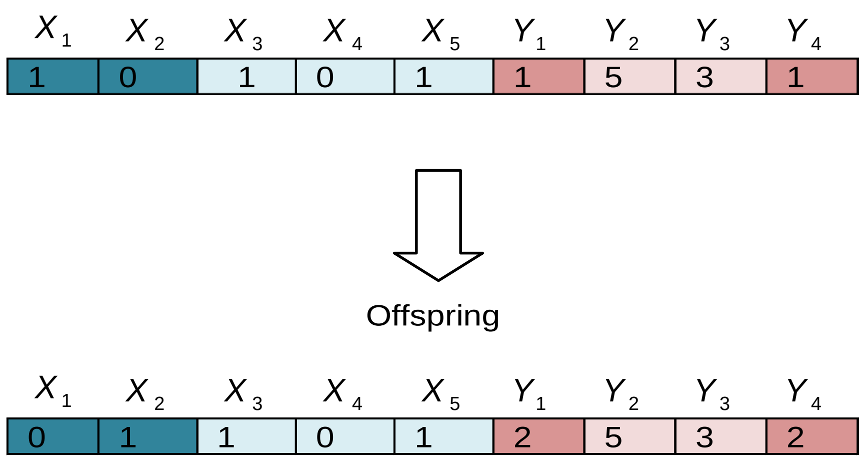

Figure 6. Sample of mutation type 2.

Pentium 5 computer with 1.00 GB RAM and 2.00 GHz CPU.

Test problems and parameter setting

Three sets of experiments were designed in order to test the performance of the proposed Lagrangian relaxation and to study the benefits of considering supply uncertainty in the model. The objective of the first experiment was to evaluate the performance of the proposed solution method in terms of the solution quality and time. The second experiment was designed to compare the performance of the presented solution approach with a popular solution heuristic in the literature, simulated annealing. The focus of the third experiment was on the benefits of considering the possibility of unreliable performances of DCs during the supply chain design phase. The experiments were implemented on the well known benchmarks in the literature which are 49, 88 and 150-node data sets described by Daskin (1995). These data sets have been very popular in the literature and have been used in a lot of research to validate the new solution approaches (Shen, 2000, 2006; Daskin et al., 2002; Jaramillo et al., 2002; Ozsen, 2004; Shen et al., 2003; Shen and Daskin, 2005; Shen and Qi, 2007; Snyder et al., 2007; Sourirajan et al., 2007, 2008; Ozsen et al., 2008; Aryanezhad et al., 2010; Qi et al., 2010). The 49-node data set represents the capitals of the lower 48 United States plus Washington, DC; the 88-node data set indicates the 50 largest cities in the 1990 U.S. census along with the 49-node data set, minus duplicates; and the 150-node data set contains the 150 largest cities in the 1990 U.S. census. In all the experiments, population data given in Daskin (1995) were divided by 1000 to be considered as the mean of demand. We set the per-unit

Table 1. Parameters for the solution approach.

Parameter	Value
Population size of GA	30
Probability of crossover in GA	0.95
Probability of mutation in GA	0.01
The number of generations in GA	400

Table 2. Computational results for 49-node problem when *e* = 10.

	P	*s*	LB	UB	Time (s)	Gap
1	5	10	239764.5	239957.9	2	0.080662
2	5	100	239775.2	240130	2	0.147972
3	10	10	534469.9	534673.6	2	0.038113
4	10	100	534477.9	534860.4	2	0.071565
5	15	10	844573.6	844788	3	0.025386
6	15	100	844581.6	844975.6	3	0.04665
7	20	10	1174674	1174894	3	0.018729
8	20	100	1174682	1175070	3	0.03303
9	25	10	1517775	1518014	4	0.015747
10	25	100	1517771	1518075	4	0.020029

cost to transport products from distribution center j to customer i, d_{ij}, to the great-circle distance between these sites. As in Daskin et al. (2002), the fixed ordering F_j and the shipping costs g_j were set to 10. Also, the variable shipping cost A_j and inventory holding cost h were set to 5 and 1 for all candidate DCs. The values of q_j, r_j, v_j, w_j, a_j and b_j, which cannot be found in the original data sets, were set randomly using the uniform distribution on (0, 1) interval.The parameters for the genetic algorithm were set based on the optimal values suggested by Grefenstette (1986). These parameters are given in Table 1.

First set of experiments

Here, the performance of the proposed solution method is tested on the 49-node, 88-node, and 150-node data sets. Fixed costs of locating DCs (f_j) were set the same as the fixed costs in Daskin (1995). To vary the difficulty of problem instances, we used different values for the parameters e and s_i. In addition to varying the lost sales costs, we tested different values for the parameter *P*. Also, we used different weights for the instances of the problem. For the 49-node data set, the weights β and θ were set to 0.1. However, for the 88 and 150-node data sets, the parameters β and θ were set to 1 and 0.0005, respectively. Tables 2 and 7 summarize the related results for the computational study on 49, 88 and 150-node with different values for the parameters P, s_i and e. In these tables the columns marked P, *s* and *e* give the parameters P, s_i and e, respectively. The columns marked Time indicate the CPU time in seconds. The columns labeled LB represent the value of lower bound, and the columns marked UB gives the value of upper bound. The last column in each table indicates the percentage gap between the obtained upper and lower bounds and it is calculated by $\frac{UB - LB}{LB} \times 100$. It follows from Tables 2 to 7 that the gap does not exceed 0.187% with different values for the parameters P, s_i, *e*, θ and β. It demonstrates that the bounds provided by the Lagrangian relaxation process are very tight and close to optimal values. Thus, the developed solution approach is able to obtain solutions close to the optimal values for the nonlinear model in a logical time.

Second set of experiments

Here, the performance of the proposed solution approach is compared with simulated annealing (SA) algorithm which has been extensively used by researchers for solving large-sized problems in the literature of facility location and supply chain design models (Chiyoshi and Galvao, 2000; Drezner et al., 2002; Wu et al., 2002; Jayaraman and Ross, 2003; Jr et al., 2006; Yigit et al.,

Table 3. Computational results for 49-node problem when e = 100.

	P	*s*	LB	UB	Time (s)	Gap
1	5	10	239767.7	240018.7	2	0.104685
2	5	100	239856.5	240305.8	2	0.18732
3	10	10	534468.3	534839.9	2	0.069527
4	10	100	534502.7	535106.8	2	0.113021
5	15	10	844569.1	845035.8	3	0.055259
6	15	100	844577.1	845292.1	3	0.084658
7	20	10	1174671	1175164	3	0.041969
8	20	100	1174679	1175378	3	0.059506
9	25	10	1517778	1518416	4	0.042035
10	25	100	1517797	1518601	4	0.052972

Table 4. Computational results for 88-node problem when e = 10.

	P	*s*	LB	UB	Time (s)	Gap
1	5	10	192799.1	192845.5	5	0.024067
2	5	100	192821.5	192906.7	5	0.044186
3	10	10	437794.6	437904.8	5	0.025172
4	10	100	437816.8	437964.9	5	0.033827
5	15	10	700892.9	701044.8	6	0.021672
6	15	100	700915.6	701104.8	6	0.026993
7	20	10	978989.5	979205.3	6	0.022043
8	20	100	979012.1	979260.5	6	0.025373
9	25	10	1273386	1273660	7	0.021517
10	25	100	1273409	1273711	7	0.023716
11	30	10	1576380	1576702	8	0.020427
12	30	100	1576403	1576751	8	0.022076
13	35	10	1886577	1886926	9	0.018499
14	35	100	1886599	1886975	9	0.01993
15	40	10	2212670	2213048	10	0.017083
16	40	100	2212692	2213095	10	0.018213

2006; Al-khedhairi, 2008, Azad and Davoudpour, 2010; Pishvaee et al., 2010). SA is a popular search algorithm capable of escaping from local optima (Henderson et al., 2003). The SA methodology draws its analogy from the annealing process of solids. In the annealing process, a solid is heated to a high temperature and gradually cooled to a low temperature to be crystallized. Since the heating process allows the atoms to move randomly, it gives the atoms enough time to align themselves in order to reach a minimum energy. This analogy can be used in combinatorial optimization in which the states of the solid correspond to the feasible solutions, the energy at each state corresponds to the improvement in the objective function and the minimum energy state will be the optimal solution (Henderson et al., 2003). Figure 7 shows the steps of the SA algorithm for the proposed model in this study, where the following parameters are used:

TI_0: The initial temperature,

CS: The rate of the current temperature decreases (cooling schedule),

ST: The freezing temperature (the temperature at which the desired energy level is reached),

L: Number of accepted solutions at each temperature,

SN: Counter for the number of accepted solutions at each temperature,

X_0: The initial solution,

X: The current solution in iterations,

X_{nh}: A solution which can be selected in the neighborhood of X in each iteration,

X_{best}: The best solution obtained in iterations, phases and algorithm,

Table 5. Computational results for 88-node problem when e = 100.

	P	***s***	**LB**	**UB**	**Time (s)**	**Gap**
1	5	10	192809.7	192927	5	0.060837
2	5	100	192832.1	193015.4	5	0.095057
3	10	10	437810.1	438392.6	5	0.133049
4	10	100	437832.3	438485.1	5	0.149098
5	15	10	700910.1	701741.5	6	0.118617
6	15	100	700932.8	701823	6	0.127002
7	20	10	979009.1	980300.1	6	0.131868
8	20	100	979031.6	980377.9	6	0.137513
9	25	10	1273415	1275120	7	0.133892
10	25	100	1273432	1275192	7	0.138209
11	30	10	1576409	1578417	8	0.127378
12	30	100	1576431	1578486	8	0.130358
13	35	10	1886609	1888747	9	0.113325
14	35	100	1886632	1888814	9	0.115656
15	40	10	2212710	2214977	10	0.102454
16	40	100	2212732	2215043	10	0.104441

Table 6. Computational results for 150-node problem when e = 10.

	P	***s***	**LB**	**UB**	**Time (s)**	**Gap**
1	5	10	499989.3	500037.4	8	0.00962
2	5	100	500038.5	500109.3	8	0.014159
3	10	10	999968.6	1000047	8	0.00784
4	10	100	1000017	1000122	8	0.0105
5	15	10	1499963	1500061	9	0.006533
6	15	100	1500013	1500134	9	0.008067
7	20	10	1999964	2000072	9	0.0054
8	20	100	2000013	2000153	9	0.007
9	25	10	2499961	2500085	10	0.00496
10	25	100	2500013	2500163	10	0.006
11	30	10	2999959	3000099	11	0.004667
12	30	100	3000013	3000176	11	0.005433
13	35	10	3499959	3500115	12	0.004457
14	35	100	3500013	3500182	12	0.004829
15	40	10	3999958	4000128	13	0.00425
16	40	100	4000020	4000241	13	0.005525
17	45	10	4499958	4500147	15	0.0042
18	45	100	4500014	4500207	15	0.004289
19	50	10	4999958	5000162	17	0.00408
20	50	100	5000014	5000230	17	0.00432
21	55	10	5499958	5500174	19	0.003927
22	55	100	5500014	5500238	19	0.004073
23	60	10	5999958	6000194	24	0.003933
24	60	100	6000014	6000259	24	0.004083

$C(X)$: The objective function value for the solution X.

In the developed SA algorithm, encoding structure in the GA is used to indicate a solution. The mutation operator defined in the previous study is applied as a neighbor generation mechanism. The approach proposed by

Table 7. Computational results for 150-node problem when *e* = 100.

	P	*s*	LB	UB	Time (s)	Gap
1	5	10	500022.8	500038.8	8	0.0032
2	5	100	500071.4	500177.6	8	0.021237
3	10	10	1000023	1000053	8	0.003
4	10	100	1000073	1000194	8	0.012099
5	15	10	1500025	1500075	9	0.003333
6	15	100	1500073	1500208	9	0.009
7	20	10	2000025	2000098	9	0.00365
8	20	100	2000073	2000240	9	0.00835
9	25	10	2500020	2500125	10	0.0042
10	25	100	2500074	2500280	10	0.00824
11	30	10	3000020	3000155	11	0.0045
12	30	100	3000074	3000309	11	0.007833
13	35	10	3500020	3500193	12	0.004943
14	35	100	3500074	3500323	12	0.007114
15	40	10	4000013	4000195	13	0.00455
16	40	100	4000074	4000374	13	0.0075
17	45	10	4500020	4500278	15	0.005733
18	45	100	4500074	4500454	15	0.008444
19	50	10	5000020	5000354	17	0.00668
20	50	100	5000074	5000484	17	0.0082
21	55	10	5500020	5500429	19	0.007436
22	55	100	5500074	5500546	19	0.008582
23	60	10	6000020	6000536	24	0.0086
24	60	100	6000074	6000636	24	0.009367

Kirkpatrick et al. (1983) is used for parameter setting of the developed SA. In other words, the initial temperature TI_0 is set to 120 in order that the probability of accepting worst solutions is at least of 80%. Typically, 0.75≤CS≤0.95, thus CS is set to 0.8. Also, the freezing temperature is set to be $ST = 0.08 \times TI_0$ (Kirkpatrick et al., 1983). Table 8 compares the results of the proposed solution approach based on Lagrangian relaxation and genetic algorithm with SA on several benchmarks.

The column labeled Problem demonstrates which benchmark is used and the column marked *P* gives the value of parameter *P*. The columns labeled Cost indicate the objective values obtained by SA and by the presented solution method, respectively. Also, the columns marked Time represent the total numbers of CPU seconds required for SA and the proposed solution approach, respectively. The last column indicates the percentage difference between the objective value obtained by SA and the objective value obtained by the proposed solution method.

In other words, the last column represents the amount of improvement in the objective value when the proposed solution approach based on Lagrangian relaxation and GA is applied instead of SA. From Table 8, it can be seen that the presented method based on Lagrangian relaxation and GA outperforms SA in both the quality of solutions and run time.

Third set of experiments

Here analyzes the benefits of considering the possibility of unreliable performances of distribution centers in the supply chain design phase. That is, the experiment was designed to imply how the total costs can be reduced when we consider the penalty costs for the unreliable performances of DCs in the supply chain design model. In addition, this experiment shows that the presence of costs of unreliable performances of DCs in the model affects the number of customers which are selected to be served.

To demonstrate the benefits of considering the possibility of unreliable performances of DCs in the model, we compared two different policies for designing the supply chain. The first policy makes the supply chain design decisions without taking the possibility of unreliable performances of DCs into consideration. In other words, the first policy determines the decision variables of the problem (X_j and Y_{ij}) with the assumption

Select an initial solution, X_0

$X_{best} = X_0$, $X = X_0$

While ($TI_0 < ST$) Do

$SN = 0$

While ($SN < L$) Do

Generate solution X_{nh} in the neighborhood of X ,

$\Delta C = C(X_{nh}) - C(X)$

If $\Delta C \leq 0$ then

$X = X_{nh}$

$SN = SN + 1$

If $C(X_{nh}) < C(X_{best})$ then

$X_{best} = X_{nh}$

End If

Else

Generate $y' \rightarrow U(0,1)$ Randomly

Set $z' = e^{-\frac{\Delta C}{TI_0}}$

If $y' < z'$ then

$X = X_{nh}$

$SN = SN + 1$

End If

End If

End While

$TI_0 = CS \times TI_0$

End While

Figure 7. Steps of the simulated annealing algorithm.

that DCs perform perfectly. The second policy, however, considers the possibility of unreliable performances of DCs when the decision variables (X_j and Y_{ij}) are determined. In fact, the second policy takes the penalty costs for unreliable performances of DCs into account, as we do in this paper. We tested these policies on the examples with 49, 88 and 150-node data sets and compared the results.

The fixed costs of locating DCs (f_j) were obtained by dividing the fixed costs in Daskin (1995) by 100. The weights β and θ were set to 1. We set $e = 1000$ to model the competitive environments, where the penalty costs for losing demand of customers due to the unreliable performances of DCs are high.

The values of other parameters were set as the same as the first set of experiments. Tables 9 to 11 present the results for the 49, 88 and 150-node data sets with different values for the parameters P and s_i . In these tables, the columns marked P and s give the parameters P and s_i , respectively. The columns marked $TC1$ represent the expected total costs when the first policy is used for making the supply chain design decisions. To derive $TC1$ for each instance of problem, first we set $e = 0$ and obtain the decisions variables (X_j and Y_{ij} for each $i \in I$ and for each $j \in J$) by solving the proposed integrated model. Then we set $e = 1000$ and calculate

Table 8. Comparison of results: SA algorithm and proposed solution approach based on Lagrangian relaxation and genetic algorithm.

	Problem	P	Simulated annealing algorithm		Proposed solution approach based on Lagrangian relaxation and genetic algorithm		
			Cost	Time (s)	Cost	Time (s)	Improvement percent in cost (%)
1	49-node	5	245105.1	10	239957.9	2	2.1
2	49-node	10	552348.8	11	534673.6	2	3.2
3	49-node	15	867338.8	14	844788	3	2.6
4	49-node	20	1220035	18	1174894	3	3.7
5	88-node	15	729495.1	29	701044.8	6	3.9
6	88-node	20	1021069	31	979205.3	6	4.1
7	88-node	25	1353518	35	1273660	7	5.9
8	88-node	30	1663188	39	1576702	8	5.2
9	88-node	35	2003106	44	1886926	9	5.8
10	88-node	40	2364368	52	2213048	10	6.4
11	150-node	10	1077637	73	1000047	8	7.2
12	150-node	20	2171631	84	2000072	9	7.9
13	150-node	30	3275217	102	3000099	11	8.4
14	150-node	40	4410284	127	4000128	13	9.3
15	150-node	45	4994614	154	4500147	15	9.9
16	150-node	50	5574317	189	5000162	17	10.3
17	150-node	55	6243103	241	5500174	19	11.9
18	150-node	60	6833934	356	6000194	24	12.2

the objective function of Equation 17 with the obtained values for the decision variables. The resulted value of the objective function can represent the expected total cost when the first policy is used. The columns labeled *N*1 indicate the number of customers which are not served when the first policy is adopted to design the supply chain.

To gain *N*1, first we set $e = 0$ and obtain the decisions variables (X_j and Y_{ij} for each $i \in I$ and for each $j \in J$) by solving the proposed integrated model. Then, the number of decision variables Y_{iz} (for each $i \in I$) which takes the value of 1 can represent the number of customers which are not served. The columns marked *TC*2 represent the expected total costs when the second policy is used for determining supply chain design decisions.

To obtain *TC2*, the integrated model is solved and the resulted objective value can represent the expected total cost. The columns labeled *N*2 indicate the number of customers which are not served if the second policy is used for supply chain design. To obtain *N*2, the proposed integrated model is solved when $e = 1000$. Then, the number of decision variables Y_{iz} which takes the value of 1 can represent the number of customers which are not served. The last column in each table implies the differences between the expected total costs of the first policy and the second one and it is calculated by:

$$\frac{TC1 - TC2}{TC1} \times 100 .$$

That is, the last columns indicate the percent of cost saving which could be provided when the second policy is used instead of the first policy for each instance of the problem. Some managerial insights can be driven from Tables 8 to 10.

First, considering the possibility of unreliable performances of DCs in the supply chain design model can lead to significant cost savings. As it can be observed from these tables, the benefit of considering such possibility in the model can be

Table 9. The benefits of considering the possibility of unreliable performances of DCs for 49-node problem.

	P	*s*	*TC*1	*N*1	*TC*2	*N*2	Cost difference (%)
1	5	20	236846.9	10	50610.59	31	78.63151925
2	5	30	509913.3	2	55286.75	6	89.1576178
3	10	20	411094	2	56044.02	26	86.36710479
4	10	30	519441.7	1	59625.77	5	88.52118072
5	15	20	550074.9	3	64700.61	20	88.23785386
6	15	30	627699.2	0	68926.72	7	89.01914876
7	20	20	492790.6	2	77701.61	18	84.23232853
8	20	30	588981.1	0	80285.05	0	86.3688235
9	25	20	617744.9	1	96212.22	12	84.42525062
10	25	30	670688.9	0	97801.52	2	85.4177524

Table 10. The benefits of considering the possibility of unreliable performances of DCs for 88-node problem.

	P	*s*	*TC*1	*N*1	*TC*2	*N*2	Cost difference (%)
1	5	20	829661.6	30	65875.09	32	92.06000518
2	5	30	919286.1	6	68792.62	20	92.51673412
3	10	20	680485.6	21	72046.57	23	89.41247746
4	10	30	759262	3	73471.71	8	90.3232732
5	15	20	738381.3	12	73467.64	31	90.05017636
6	15	30	1095399	1	74861.83	13	93.16579547
7	20	20	1130649	18	77548.3	21	93.14125691
8	20	30	1231921	0	81308.14	13	93.39988986
9	25	20	740021.4	20	85887.38	26	88.39393264
10	25	30	957896	1	86140.96	8	91.00727389
11	30	20	885314.5	19	97281.69	23	89.01162336
12	30	30	969269.6	1	100851.5	4	89.59510325
13	35	20	817931.6	9	113336	17	86.14358546
14	35	30	1097820	0	114100.3	6	89.60665097
15	40	20	950010.7	11	133433.3	15	85.95454822
16	40	30	972183.3	3	133574.4	8	86.26036862

be significant up to 94%.

Also it can be seen from these tables that the amount of cost saving for a problem with $s = 30$ is more than for the same problem with $s = 20$. This can show that the amount of cost saving increases as the value of s increases. In addition, for each instance of the problem we have $N1 < N2$. In other words, when the possibility of unreliable performances of DCs is considered in the design phase, fewer customers are selected to be served.

The reason can be due to the fact that when such possibility is considered, the model decreases the selected customers in order to reduce the risk of incurring penalty costs for unreliable performances of DCs.

Conclusion

This paper has investigated the design of supply chain with random demands where the supplier and distribution centers are unreliable. When the cost of serving the customers is prohibitive the supply chain may choose not to serve them at all.

An integrated supply chain design model has been presented that simultaneously determines which customers need to be served, where distribution centers are located, which distribution centers are assigned to which customers and how much products are ordered to the supplier by each DC.

The model has been formulated as a nonlinear integer programming that minimizes the expected total cost including costs of location, inventory, transportation and lost sales. A solution approach based on Lagrangian relaxation and genetic algorithm has been developed which is capable of solving the problem effectively. Besides, we have conducted numerical experiments to show that significant cost savings can be achieved if we consider the possibility of unreliable performances of DCs

Table 11. The benefits of considering the possibility of unreliable performances of DCs for 150-node problem.

	P	*s*	*TC*1	*N*1	*TC*2	*N*2	Cost difference (%)
1	5	20	1115864	61	82019.4	73	92.64969618
2	5	30	1891929	13	91373.48	55	95.17035293
3	10	20	1279921	56	89359.7	79	93.0183451
4	10	30	1699675	7	100102.5	57	94.11049114
5	15	20	1799409	28	96594.57	71	94.63187186
6	15	30	1894272	2	100635.5	43	94.68737827
7	20	20	1132547	43	102159.6	79	90.97965595
8	20	30	1724173	4	113612.6	49	93.41060156
9	25	20	1358116	34	111097.8	77	91.81971385
10	25	30	1719241	2	116167.9	33	93.24306941
11	30	20	1517994	19	119740.3	71	92.11194095
12	30	30	1899941	1	127121.7	50	93.30917578
13	35	20	1667698	14	131390.4	66	92.12144907
14	35	30	1971664	1	145262.9	40	92.63247145
15	40	20	1580987	15	143867.2	77	90.90016805
16	40	30	1680123	1	147808.7	35	91.20250908
17	45	20	1534083	9	157183.6	60	89.75390541
18	45	30	1602666	0	161595.4	29	89.9170885
19	50	20	1583160	4	172870.4	58	89.08067399
20	50	30	1642688	0	177994.5	40	89.16443869
21	55	20	1732884	1	187854.8	41	89.15941257
22	55	30	1831674	0	190892.2	23	89.57826601
23	60	20	1480603	5	169438.9	61	88.55608709
24	60	30	1770346	1	175139.1	25	90.10706899

in the supply chain design phase.

This work can be extended in some directions. For instance, it would be an interesting area for future research to extend the proposed model for capacitated facilities and multiple products.

Also, the model can be extended to include the routing decisions. In addition, the model will be more useful in the real world, when the customers can be served by multiple distribution centers.

REFERENCES

Al-khedhairi A (2008). Simulated Annealing Metaheuristic for Solving P-Median Problem. Int. J. Contemp. Math. Sci., 3(28): 1357-1365.

Alp O, Erkut E, Drezner Z (2003). An efficient genetic algorithm for the p-median problem. Anna. Oper. Res., 122: 21-42.

Aryanezhad MB, Jalali SG, Jabbarzadeh A (2010). An integrated supply chain design model with random disruptions consideration. Afr. J. Bus. Manage., 4(12): 2393-2401.

Azad N, Davoudpour H (2010). Designing a Reliable Supply Chain Network Model under Disruption Risks. Am. J. Sci., 6(12): 1091-1097.

Baten A, Kamil AA (2009). Analysis of inventory-production systems with Weibull distributed deterioration. Int. J. Phys. Sci., 4(11): 676-682.

Barrionuevo A, Deutsch CH (2005). A Distribution System Brought to Its Knees. New York Times. p. C1, Sep. 1.

Berman O, Krass D, Menezes MBC (2007). Facility reliability issues in network p-median problems: strategic centralization and co-location effects. Oper. Res., 55(1): 332-350.

Chen KK, Chiu YSP, Hwang MH (2010). Integrating a cost reduction delivery policy into an imperfect production system with repairable items. Int. J. Phys. Sci., 5(13): 2030-2037.

Cheng FT, Ting CK (2010). Determining economic lot size and number of deliveries for EPQ model with quality assurance using algebraic approach. Int. J. Phys. Sci., 5(15): 2346-2350.

Chiyoshi F, Galvao RD (2000). A statistical analysis of simulated annealing applied to the p-median problem. Ann. Oper. Res., 96: 61-74.

Church RL, Scaparra MP (2007). Protecting critical assets: the r-interdiction median problem with fortification. Geographical. Anal., 39: 129-146.

Cui T, Ouyang Y, Shen ZJM (2010). Reliable facility location design under the risk of disruptions. Oper. Res., 58(4): 998-1011.

Current J, Daskin M, Schilling D (2001). Discrete Network Location Models. In Drezner Z, Hamacher HW (eds) Facility Location: Applications and Theory, Springer-Verlag, Berlin, pp. 83-120.

Daskin MS (1995). Network and discrete location: models, algorithms, and applications. Wiley. New York.

Daskin MS, Coullard C, Shen ZJ (2002). An inventory-location model: formulation, solution algorithm and computational results. Anna. Oper. Res., 110(1): 83-106.

Drezner T, Drezner Z, Salhi S (2002). Solving the multiple competitive facilities location problem. Eur. J. Oper. Res., 142: 138-151.

Drezner Z, Wesolowsky G (2003). Network design: selection and design of links and facility location. Transp. Res. Part A. 37: 241-256.

Erdem AS, Ozekici S (2002). Inventory models with random yield in a random environment. Int. J. Prod. Econ., 78: 239-253.

Erlebacher S J, Meller RD (2000). The interaction of location and inventory in designing distribution systems. IIE, 32(1): 155 – 166.

Fisher ML (1981). The Lagrangian relaxation method for solving integer programming problems. Manage. Sci., 27: 1-18.

Fisher ML (1985). An applications oriented guide to Lagrangian

relaxation. Interfaces, 15: 2-21.
Garey MR, Johnson DS (1979). Computers and Intractability: A Guide to the Theory of NP-Completeness WH, Freeman and Co., New York.
Gen M, Cheng R (1996). Genetic algorithms and engineering design. Wiley. New York.
Gen M, Cheng R (2000). Genetic algorithms and engineering optimization. Wiley. New York.
Goldberg DE (1989). Genetic Algorithms in Search, Optimization and Machine Learning. Addison-Wesley.
Grefenstette JJ (1986). Optimization of control parameters for genetic algorithms. IEEE Trans. Systems, Man, Cybern., 16(1): 122-128.
Henderson D, Jacobson SH, Johnson AW (2003). The theory and practice of simulated annealing. In Glover et al. (eds) Handbook of Metaheuristics, Kluwer Academic Publishers, New York, pp, 287-320.
Jaramillo JH, Bhadury J, Batta R (2002). On the use of genetic algorithms to solve location problems. Comput. Oper. Res., 29: 761-779.
Jayaraman V, Ross A (2003). A simulated annealing methodology to distribution network design and management. Eur. J. Oper. Res., 144: 629-645.
Jr MAA, Kadipasaoglu SN, Khumawala BM (2006). An empirical comparison of Tabu Search, Simulated Annealing, and Genetic Algorithms for facilities location problems. Int. J. Prod. Econ., 103: 742-754.
Kirkpatrick S, Gelatt CD, Vecchi MP (1983). Optimization by simulated annealing. Science, 220: 45-98.
Lim M, Daskin MS, Bassamboo A, Chopra S (2009). A facility reliability problem: Formulation, properties and algorithm. Naval Res. Logist., 57(1): 58-70.
Melo M, Nickel S, Saldanha-da-Gama F (2009). Facility location and supply chain management – A review. Eur. J. Oper. Res., 196: 401-412.
Min H, Ko HJ, Ko, CS (2006). A genetic algorithm approach to developing the multi-echelon reverse logistics network for product returns. Omega, 34: 56-69.
Ozsen L (2004). Location-inventory planning models: Capacity issues and solution algorithms. PhD Dissertation, Northwestern University, Illinois, USA.
Ozsen L, Coullard CR, Daskin MS (2008). Capacitated warehouse location model with risk pooling. Naval Res. Logist., 55(4) (1): 295–312.
Ozsen L, Daskin MS, Coullard CR (2009). Facility Location Modeling and Inventory Management with Multisourcing. Transp. Sci., 43: 455-472.
Pishvaee MS, Kianfar K, Karimi B (2010). Reverse logistics network design using simulated annealing. Int. J. Adv. Manuf. Technol., 47:269–281.
Qi L, Shen ZJ (2007). A supply chain design model with unreliable supply. Naval Res. Logist., 54: 829-844.
Qi L, Shen ZJ, Snyder LV (2009). A continuous-review inventory model with disruptions at both supplier and retailer. Prod. Oper. Res. Manage., 18: 516-532.
Qi L, Shen ZJ, Snyder LV (2010). The effect of supply disruptions on supply chain design decisions. Transp. Sci., 44: 274-289.
Radjou N (2002). Adapting to Supply Network Change. Forrester Research Tech Strategy Report. Forrester Research. Cambridge. MA.
Ross SM (2007). Introduction to Probability Models. Academic Press.
Scaparra MP, Church RL (2008). A bilevel mixed-Integer program for critical infrastructure protection planning. Comput. Oper. Res., 35(6): 1905-1923.
Shen ZJ (2000). Efficient algorithms for various supply chain problems. PhD Dissertation, Northwestern University, Illinois, USA.
Shen ZJ (2006). A Profit Maximizing Supply Chain Network Design Model with Demand Choice Flexibility. Operations. Res. Lett., 34: 673-682.
Shen ZJ (2007). Integrated supply chain design models: a survey and future research direction. J. Ind. Manage. Optim., 3(1): 1-27.
Shen ZJ, Coullard C, Daskin MS (2003). A joint location-inventory model. Transp. Sci., 37(1): 40-55.
Shen ZJ, Daskin M (2005). Trade-offs Between Customer Service and Cost in Integrated Supply Chain Design. M&SOM, 7(1): 188-207.
Shen ZJ, Qi L (2007). Incorporating inventory and routing costs in strategic location models. Eur. J. Oper. Res., 179(1): 372-389.
Shu J, Teo CP, Shen ZJ (2005). Stochastic transportation-inventory network design problem. Oper. Res., 53(1): 48-60.
Snyder LV, Daskin MS (2005). Reliability models for facility location: the expected failure cost case. Transp. Sci., 39(1): 400-416.
Snyder L, Daskin M (2006). A random-key genetic algorithm for the generalized traveling salesman problem. Eur. J. Oper. Res., 174: 38-53.
Snyder LV, Daskin MS (2007). Models for reliable supply chain network design. In Murray A and Grubesic TH (eds) Reliability and Vulnerability in Critical Infrastructure: A Quantitative Geographic Perspective, Springer, Germany, pp 257-289.
Snyder LV, Daskin MS, Teo CP (2007). The stochastic location model with risk pooling. Eur. J. Oper. Res., 179(1): 1221-1238.
Snyder LV, Scaparra MP, Daskin MS, Church RL (2006). Planning for disruptions in supply chain networks. In Johnson MP, Norman B, Secomandi N (eds) Tutorials in Operations Research. Informs, Baltimore, 9: 234-257.
Sourirajan K, Ozsen L, Uzsoy R (2007). A single product network design model with lead time and safety stock considerations. IIE. Trans., 39: 411-424.
Sourirajan K, Ozsen L, Uzsoy R (2008). A genetic algorithm for a single product network design model with lead time and safety stock considerations. Eur. J. Oper. Res., 197: 599-608.
Wazed MA, Ahmed S, Nukman Y (2010a). Impacts of quality and processing time uncertainties in multistage production system. Int. J. Phys. Sci., 5(6): 814-825.
Wazed MA, Ahmed S, Yusoff N (2010b). Impacts of common processes in multistage production system under machine breakdown and quality uncertainties. Afr. J. Bus. Manage., 4(6): 979-986.
Wu TH, Low C, Bai JW (2002). Heuristic solutions to multi-depot location-routing problems. Comput. Oper. Res., 29: 1393-1415.
Wu X (2008). (Q; r) Inventory Policies under Uncertain Supply Chain Environment. PhD Dissertation, North Carolina State University, Raleigh, USA.
Yigit Y, Aydin ME, Turkbey O (2006). Solving large-scale uncapacitated facility location problems with evolutionary simulated annealing. Int. J. Prod. Res., 44(22): 4773-4791.