Queuing approach and optimal inventory decisions in a stochastic supply chain network design

Ehsan Dehghani¹,*, Peyman Taki¹

Abstract

This paper addresses an integrated multi-echelon location-allocation-inventory problem in a stochastic supply chain. In a bid to be more realistic, the demand and lead time are considered to be hemmed in by uncertainty. To tackle the proposed supply chain network design problem, a two-phase approach based on queuing and optimization models is devised. The queuing approach is first deployed, which is able to cope with inherent uncertainty of parameters. Afterwards, the proposed supply chain network design problem is formulated using a mixed-integer nonlinear model. Likewise, the convexity of the model is proved and the optimal inventory policy as closed-form is acquired. Inasmuch as the concerned problem belongs to NP-hard problems, two meta-heuristic algorithms are employed, which are capable of circumventing the complexity burden of the model. The numerical examples evince the efficient and effective performance of the solving algorithms. Lastly, sensitivity analyses are conducted through which interesting insights are gained.

Keywords: Supply chain; Stochastic modeling; Queuing theory; Inventory theory; Operations research.

Received: February 2019-09

Revised: May 2019-25

Accepted: August 2019-05

1. Introduction

A supply chain (SC) is a system comprising activities, people, organizations, resources, and information which is involved services or moving a product from manufacturers to customers (Nagurney, 2006). The competitive conditions of today’s world eventuate in the companies focus on improving their SC performance in order to reduce costs and increase the customer satisfaction (Diabat and Theodorou, 2015). One of the most prominent issues in logistics and SC is designing the integrated multi-echelon distribution systems. That is, considering strategic, tactical and operational decisions in an incorporated manner can ameliorate the management across the SC and yield a full-optimized network (Miranda and Garrido, 2009; Sadjadi et al., 2016). Facility location and inventory management problems are the two most outstanding problems in SC optimization that have salient impacts on efficient design of SC networks (Gunasekaran et al., 2001; Stevens, 1989).

* Corresponding author; ehsandehghan@alumni.iust.ac.ir

¹ School of Industrial Engineering, Iran University of Science and Technology, Tehran, Iran.
Facility location problems, as strategic decisions, encompass determining the optimal number of open plants and their locations. On the other side, inventory management problems identify the responsiveness of systems in terms of stock availability as tactical decisions. Now, in view of pervasive speaking, considering the integrated model of both decisions can lead to better solutions (Diabat, Battaïa, et al., 2015; Ramezani and Naderi, 2018).

In this research, we have presented a joint location-inventory problem in a three-level SC, including a single manufacturer, multiple potential distribution centers (DCs) and retailers. The aim is to identify a set of open DCs to meet the demands coming from retailers. The objective of the presented model is to minimize the total costs which is composed of the fixed cost of locating DCs, transportation cost from DCs to retailers and inventory costs. To make the problem more similar to real-life problems, the demand and lead-time are assumed to be hemmed in by uncertainty. Additionally, there is facility capacitation restriction. Accordingly, our model is an extension of capacitated facility location model, known as a NP-hard problem (Mirchandani and Francis, 1990). To circumvent the complexity burden of the model, two meta-heuristic algorithms, i.e. simulated algorithm (SA) and genetic algorithm (GA), have been devised to solve the problem, especially for large instances to reach near optimal solutions in a logical CPU time. In accordance with the best of our knowledge and the literature review, the main contributions of this research can be considered as follows. 1) This research is one the first studies in the field of location-inventory problems that cope with uncertainty with regard to queuing approach. 2) We have obtained a closed-form for optimal inventory policy of DCs by proving the convexity of objective function. Optimal control policies for stochastic inventories are the contribution of some researches such as Federgruen and Zheng (1993), Chao et al. (2012) and Chen and Feng (2006) in the field of inventory control.

Each open DC holds inventory and replenishes its inventory by ordering to the manufacturer under (S-I, S) policy. In current policy, the peak level of stock is S (i.e., base stock level) and an order will be placed, whenever the stock level (the summation of on hand and on order quantities) decreases to S-1. Speaking intuitively, an order will be placed, whenever a demand is received and satisfied (Hill, 1999). Some applications of (S-I, S) inventory policy are as follows; 1) military, airlines and computer industries, 2) the case of unit size and recoverable-item inventory systems, 3) the products with expensive stock level controlling, low demand and high holding cost (Kalpakam and Shanthi, 2001), and 4) for spare parts system, when a failed part is exchanged by a new one. Likewise, if the failure rate is low, this inventory policy will be more desirable (Schultz, 1990).

The organization of this paper is as follows. The literature of location-inventory models considering queuing theory approaches are presented in Section 2. Section 3 elaborates the model description and mathematical formulation of the concerned problem. In Section 4, solution approach to solve the proposed problem is surveyed. In Section 5, the computational results and sensitivity analysis are carried out. Lastly, some concluding remarks are presented in Section 6.

2. Literature Review

Here, the literature review has been categorized into two sub-sections comprising location-inventory problems and inventory control models with queuing theory approaches.

2.1. Location-inventory problem

Baumol and Wolfe (1958) are the first authors who considered integrated inventory with location costs. They have studied a procedure to locate the variable number of warehouses. Two-level facility location where the second level has limited capacity has been considered by
Tragantalerngsak et al. (2000). Daskin et al. (2002) have proposed a location-inventory problem considering deterministic lead-time. They also have implemented a number of heuristics to finding near optimal solutions. A routing-inventory network design problem considering risk-pooling has been considered by Shu et al. (2005). This research is the developed model of Shen et al. (2003), where two different algorithms to solve the general model have been proposed. Chew et al. (2007) have studied a stochastic integrated location-inventory model aimed to minimize the total costs including the transportation, inventory and ordering costs in multiple capacitated distribution centers. They have used GA to solve their problem. A capacitated integrated location-inventory model with risk pooling has been investigated by Ozen et al. (2009). They have used normal distribution function to approximate Poisson demand process. Park et al. (2010) have presented a stochastic network design problem with regard to economic order quantity (EOQ) to approximate the optimal solution of the \((r, Q)\) inventory policy. Ahmadi-Javid and Seddighi (2012) have formulated a stochastic distribution network design problem in a SC as a mixed integer convex programming model, which minimizes the total costs of location, inventory and delivery delay. They have developed a heuristic solution approach based on SA and Tuba search to solve large-sized problems. An integrated location-inventory model has been studied by Berman et al. (2012). They have assumed each open DC has a periodic-review and they have modeled their problem as a nonlinear integer-programming problem and used an approximate procedure for determining inventory policy. Diabat, Abdallah, et al. (2015) have provided a closed-loop location-inventory model as a mixed integer nonlinear programming, where the lead-time is deterministic. Shahhabi et al. (2014) have developed a stochastic SC considering the correlated demands across the retailers. They have developed a solution approach based on an outer approximation strategy and they have illustrated that their solution procedure performs effective and efficient. Jindal, and Solanki (2016) have presented a two single-vendor single-buyer integrated SC inventory model with inflation and time value of money. Rabbani et al. (2016) have considered a basic mixed-integer non-linear programming model to lease new products. A three-level SC network has been introduced by Sadjadi et al. (2016), where the location, allocation and inventory replenishment decisions are simultaneously optimized. They have analyzed the inventory systems by a Markov process and then the location-inventory problem has been formulated based on obtained results. AmalNick and Qorbani (2017) have assumed dynamic pricing strategy based on demand value. They have combined neural network and evolutionary algorithms to optimize pricing policies. Applying a queuing approach, Rabbani et al. (2017) have considered a reliable location – inventory problem for a supply chain system. Last but not least, a mixed-integer linear programming model for an integrated SC has been proposed by Alshamsi and Diabat (2018). Díaz-Mateus et al. (2018) have used the constrained multinomial logit for discrete choices to estimate demand levels for a non-linear optimization model. To solve the presented model, they have implemented a metaheuristic approach based on particle swarm optimization.

### 2.2. Inventory control models considering queuing theory approaches

In this subsection due to the use of queuing approach in our research, the literature of inventory control models considering queuing theory has been reviewed. One of the earliest researches in this field has been proposed by Sigman and Simchi-Levi (1992). They have proposed a light traffic heuristic for the M/G/1 queue system. Berman and Kim (2001) have addressed an optimal inventory control, in which the demand and lead-time are uncertain with Poisson and exponential distributions, respectively. They have applied a Markov decision problem to determine the replenishment policy and then they have proposed a numerical study to evaluate the optimal performance. An integrated queuing-inventory model has been investigated by Schwarz and Daduna (2006), where lead time is exponentially distributed. They have derived
explicit performance measures for M/M/1 systems considering inventory under continuous review and backordering. Applying a queuing approach, Krishnamoorthy et al. (2013) have investigated \((s, Q)\) and \((s, S)\) inventory policies and they have analyzed numerically the per unit time expected cost for these systems. Baek and Moon (2014) have extended a queuing system incorporated with a production-inventory system and they have assumed the arrival of customers with a Poisson processes. Table 1 shows the specifications of the related literature reviews in the field of location-inventory. In this research, we have proposed a joint stochastic location-inventory problem in a three-level SC. A queuing approach has been implemented to cope with inherent uncertainty and to study the characteristics of the inventory policy. With respect to traditional inventory models, there are enormous advantages when inventory systems are described as a Markov process (see, Frizelle and Jaber (2009) and Saffari et al. (2013)). Owing to NP-hardness of the proposed problem, commercial solvers such as BARON are not able to solve the model efficiently. Accordingly, two meta-heuristic algorithms have been proposed to solve the model. Also, we could achieve the optimal inventory policy for open DCs as a closed-form with regard to the convexity of the objective function. In accordance with the literature review and Table 1, our main contributions for this research can be considered as follows. 1) Modeling queuing approach in location-inventory problems with stochastic demands and lead time. 2) Introducing a closed-form optimal inventory control for a stochastic supply chain network design problem. In this manner, the number of reorder, the mean of inventory level, backlogged demands and the base stock level are obtained. It is worthy to note that this procedure in addition to determining the optimal inventory of DCs, it can enhance the performance of the proposed algorithms in both terms of solutions’ quality and CPU Time.
### Table 1. Specification of the researches in the field of location-inventory

<table>
<thead>
<tr>
<th>Reference</th>
<th>Type of modeling</th>
<th>Objective function</th>
<th>Demand distribution</th>
<th>Lead-time distribution</th>
<th>Inventory Policy</th>
<th>Identifying inventory policy</th>
<th>Type of shortage</th>
<th>Solution procedure</th>
</tr>
</thead>
<tbody>
<tr>
<td>Daskin et al. (2002)</td>
<td>MINLP</td>
<td>Single</td>
<td>NO</td>
<td>Det.</td>
<td>(R, Q)</td>
<td>AP</td>
<td>BA</td>
<td>LA</td>
</tr>
<tr>
<td>Shu et al. (2005)</td>
<td>MINLP</td>
<td>Single</td>
<td>NO</td>
<td>Det.</td>
<td>(R, Q)</td>
<td>AP</td>
<td>BA</td>
<td>LA</td>
</tr>
<tr>
<td>Ozsen et al. (2008)</td>
<td>MINLP</td>
<td>Single</td>
<td>NO</td>
<td>Det.</td>
<td>(R, Q)</td>
<td>AP</td>
<td>BA</td>
<td>LA</td>
</tr>
<tr>
<td>Ahmadi Javid and Azad (2010)</td>
<td>MINLP</td>
<td>Single</td>
<td>NO</td>
<td>Det.</td>
<td>(R, Q)</td>
<td>AP</td>
<td>BA</td>
<td>ME</td>
</tr>
<tr>
<td>Berman et al. (2012)</td>
<td>MINLP</td>
<td>Single</td>
<td>NO</td>
<td>Det.</td>
<td>(R, T)</td>
<td>AP</td>
<td>BA</td>
<td>LA</td>
</tr>
<tr>
<td>Nekooeghadiri et al. (2014a)</td>
<td>MINLP</td>
<td>Multi</td>
<td>NO</td>
<td>Det.</td>
<td>(R, Q)</td>
<td>AP</td>
<td>BA</td>
<td>ME</td>
</tr>
<tr>
<td>Sadjadi et al. (2016)</td>
<td>MINLP</td>
<td>Single</td>
<td>PO</td>
<td>Det.</td>
<td>(S-1, S)</td>
<td>-</td>
<td>BA</td>
<td>CO</td>
</tr>
<tr>
<td>Taki et al. (2016)</td>
<td>MINLP</td>
<td>Multi</td>
<td>NO</td>
<td>NO</td>
<td>(R, Q)</td>
<td>AP</td>
<td>LO</td>
<td>ME</td>
</tr>
<tr>
<td>Rabhani et al. (2016)</td>
<td>MINLP</td>
<td>Single</td>
<td>NO</td>
<td>Det.</td>
<td>(R, Q)</td>
<td>AP</td>
<td>BA</td>
<td>ME</td>
</tr>
<tr>
<td>Jindal and Solanki (2016)</td>
<td>MINLP</td>
<td>Single</td>
<td>NO</td>
<td>NO</td>
<td>(R, Q)</td>
<td>AP</td>
<td>BA</td>
<td>LA</td>
</tr>
<tr>
<td>AmalNick and Qorbanian (2017)</td>
<td>MINLP</td>
<td>Single</td>
<td>NO</td>
<td>Det.</td>
<td>(R, Q)</td>
<td>AP</td>
<td>BA</td>
<td>ME</td>
</tr>
<tr>
<td>Díaz-Mateus et al. (2018)</td>
<td>MINLP</td>
<td>Single</td>
<td>NO</td>
<td>Det.</td>
<td>(R, Q)</td>
<td>AP</td>
<td>BA</td>
<td>ME</td>
</tr>
<tr>
<td><strong>This study</strong></td>
<td>MINLP</td>
<td>Single</td>
<td>PO</td>
<td>EX</td>
<td>(S-1, S)</td>
<td>OPT</td>
<td>BA</td>
<td>ME</td>
</tr>
</tbody>
</table>

### 3. Model Development

#### 3.1. Problem and notations definition

A multi-echelon SC distribution system including one manufacturer, multiple DCs and multiple retailers is considered. Figure 1 shows the configuration of the proposed supply chain. The demand of retailers is stochastic with Poisson probability distribution function with regard to be independent of retailers’ demand from each other. The (S-1, S) inventory policy is taken into account for DCs. Unsatisfied demand will be considered as backordered. The input demand rate is equal to \( \lambda \) that is summation of Poisson rates of allocated retailers. To response the delivered demands those ones that received earlier have priority rather the older ones. An exponentially probability distribution function with the parameter \( \mu \) has been considered for lead-time of manufacturers. It is assumed to continue serving the demands \( \lambda \leq \mu \).

In this paper, a set of integrated strategic and tactical decisions are determined in order to minimize the total costs of the supply chain network design as follows.
• Determining the number of open DCs.
• Identifying the location of open DCs from a predefined set of potential locations.
• Allocating the retailers to open DCs by considering single-sourcing concept.
• Specifying the inventory policy of each open DC.

![Figure 1. The configuration of the proposed supply chain](image)

In the following, the notations of our proposed model are introduced.

**Sets**

- $K$: Index for the set of potential DCs
- $I$: Index for the set of retailers

**Parameters**

- $C_k$: Per unit purchase cost for each product from the manufacturer ($\forall k \in K$)
- $\pi_k$: Per unit shortage cost of each product for DC $k$ ($\forall k \in K$)
- $F_k$: Fixed cost of opening DC $k$
- $T_{ki}$: Unit transportation cost from DC $k$ to retailer $i$ ($\forall k \in K$) and ($\forall i \in I$)
- $A_k$: Unit order cost for DC $k$ ($\forall k \in K$)
- $h_k$: Unit holding cost for DC $k$ ($\forall k \in K$)
- $\lambda_i$: Mean of annual demand for retailer $i$ ($\forall i \in I$)
- $\mu$: The stochastic lead-time for the manufacturer

**Decision variables**

- $z_k$: 1 if DC $k$ is open, 0 otherwise ($\forall k \in K$)
- $y_{ki}$: 1 if retailer $i$ is assigned to DC $k$, 0 otherwise ($\forall k \in K$) ($\forall i \in I$)
- $S_k$: Base stock level at DC $k$ ($\forall k \in K$)
- $\lambda_k$: Mean of annual demand at DC $k$ ($\forall k \in K$)
$OR_k$: The expected value of reorders at DC $k$ ($\forall k \in K$)

$MI_k$: The expected inventory level at DC $k$ ($\forall k \in K$)

$SH_K$: The expected value of backlogged demands at DC $k$ ($\forall k \in K$)

Using queuing approach in the proposed location-inventory model, the inventory policy has been characterized in sub-section 3.2.

### 3.2. Inventory policy for each open DC

To determine the inventory policy for each open DC, the number of reorders, the mean of inventory level and backlogged demands are obtained as follows.

Let assume $X_k(t)$ denotes the level of inventory at time $t \geq 0$ for DC $k$. The process $\{X_k(t); t \geq 0\}$ is a Markov process on the state space $\Phi_k = \{S_k, S_k - 1, ..., 0, -1, -2, -3, ...\}$. Thus, the infinitesimal generator matrix is as Eq. (3.1).

$$Q = \begin{bmatrix}
\lambda_k & -\mu \\
-\lambda_k & \lambda_k + \mu & -\mu \\
& \ddots & \ddots \\
& & \ddots & -\mu & \lambda_k + \mu & -\mu \\
& & & \ddots & \ddots & \ddots & -\mu \\
& & & & \ddots & \ddots & \ddots & \ddots & -\mu \\
& & & & & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & -\mu \\
& & & & & & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & -\mu \\
& & & & & & & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & -\mu \\
& & & & & & & & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & -\mu \\
& & & & & & & & & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & -\mu \\
\end{bmatrix}$$  \hspace{1cm} (3.1)

Also, let to consider that $\pi = (\pi_{S_k}, \pi_{S_k - 1}, \pi_{S_k - 2}, ... \pi_{j_k}, ...)$ denotes the stationary probability vector for the state space components. Then, Eq. (3.2) must be satisfied by the stationary probability vector $\pi$.

$$\pi Q = 0$$  \hspace{1cm} (3.2)

Meanwhile, regarding to that $\sum_{j_k \in \phi_k} \pi_{j_k} \pi_{j_k} = 1$, the stationary probabilities can be computed by Eqs. (3.3) and (3.4).

$$\pi_{j_k} = \left(1 - \frac{\lambda_k}{\mu}\right) \left(\frac{\lambda_k}{\mu}\right)^{S_k - j_k}$$ \hspace{1cm} \text{for } j \leq S_k - 1 \hspace{1cm} (3.3)

$$\pi_{S_k} = 1 - \frac{\lambda_k}{\mu}$$  \hspace{1cm} (3.4)

Given the $(S-1, S)$ inventory policy and the shortage of demands are backlogged, the number of reorders can be obtained by Eq. (3.5).

$$OR_k = \lambda_k$$  \hspace{1cm} (3.5)

The steady state of expected inventory level is determined by Eqs. (3.6) and (3.7).

$$MI_k = \sum_{j=0}^{S_k} j \times \pi_{j_k}$$  \hspace{1cm} (3.6)
\[ Ml_k = \frac{\left( \lambda_k S_k^{s_k+1} - \lambda_k \mu S_k - \lambda_k S_k \mu S_k + S_k \mu S_k^{s_k+1} \right) \mu \left( \frac{\lambda_k}{\mu} \right)^{S_k} \left( -1 + \frac{\lambda_k}{\mu} \right)}{\lambda_k^{S_k} (\lambda_k - \mu)^2} \]  

(3.7)

Besides, the retailers may confront shortage when the level of inventory is zero or smaller. Thereupon, the value of expected backlogged demands in each open DC is given by Eq. (3.8).

\[ SH_K = \lambda_k \sum_{j_k \leq 0} \pi_{j_k} = \lambda_k \left( \frac{\lambda_k}{\mu} \right)^{S_k} \]  

(3.8)

### 3.3. Mathematical formulation

According to the defined notations, our proposed location-inventory problem can be formulated as follows.

\[
\begin{align*}
\text{Min} & = \sum_{k \in K} F_k z_k + \sum_{k \in K} \sum_{i \in I} T_{ki} y_{ki} \lambda_i' \\
& + \sum_{k} h_k z_k \left[ \mu \left( \frac{\lambda_k}{\mu} \right)^{S_k} \left( -1 + \frac{\lambda_k}{\mu} \right) \left( \lambda_k S_k^{s_k+1} - \lambda_k \mu S_k - \lambda_k S_k \mu S_k + S_k \mu S_k^{s_k+1} \right) \right] \left( \frac{1}{\lambda_k^{S_k} (\lambda_k - \mu)^2} \right) \\
& + \pi_k z_k \left[ \lambda_k \left( \frac{\lambda_k}{\mu} \right)^{S_k} \right] + (A_k + C_k) z_k \lambda_k \\
\end{align*}
\]

(3.9)

\[
\sum_{k} y_{ki} = 1 \quad \forall i \in I
\]

(3.10)

\[
y_{ki} \leq z_k \quad \forall i \in I, \forall k \in K
\]

(3.11)

\[
\sum_{i} \lambda_i' y_{ki} = \lambda_k \quad \forall k \in K
\]

(3.12)

\[
\lambda_k \leq \mu \quad \forall k \in K
\]

(3.13)

\[
y_{ki}, z_k \in \{0,1\} \quad \forall i \in I, \forall k \in K
\]

(3.14)

\[
S_k \in \{0,1,\ldots\} \quad \forall k \in K
\]

(3.15)

The Eq. (3.9) as the objective function minimizes the costs. The first term computes the fixed cost of opening DCs. The second term shows the transportation cost from DCs to the retailers and the last term illustrates the expected inventory costs including holding, shortage, ordering and purchase costs.

Eq. (3.10) shows single-sourcing for retailers. Eq. (3.11) demonstrates the assignment constraint for DCs and retailers and provides the opening condition for DCs. Eq. (3.12) shows the demand rate of each DC which is obtained by the summation of demand rate of assigned retailers. Eq. (3.13) guarantees the stability of the inventory system to avoid the shortage. Eqs. (3.14) and (3.15) are binary and integer variables, respectively.

### 4. Solution Approach

Our proposed model is defined as well-known capacitated facility location problems which are known as NP-hard problems (Mirchandani and Francis, 1990). Therefore, exact solution...
methods are mainly inefficient to solve the proposed problem. Thus, to solve our model, two meta-heuristic algorithms which have been successfully used to tackle complex models in the literature of integrated location-inventory problems are implemented (e.g., Ahmadi Javid and Azad (2010) and Nekooghadirli et al. (2014)). Here, the convexity of the objective function is proved in Proposition 1 and then the optimal value of the base stock level is calculated as a closed-form expression in Proposition 2.

**Proposition 1.** The objective function of our proposed model is strictly convex considering the base stock level variables are continues.

**Proof.** The linear parts of the objective function are convex. Therefore, only the convexity of nonlinear parts of the objective function needs to be proved. The nonlinear parts are presented by Eq. (4.1).

\[
g(S_1, S_2, \ldots, S_n) = \sum_k h_k z_k \left( -1 + \frac{\lambda_k}{\mu} \right) \left( \lambda_k S_k^{s_k+1} - \lambda_k \mu S_k - \lambda_k S_k \mu S_k + S_k \mu S_k^{s_k+1} \right) + \pi_k z_k \left[ \lambda_k \left( \frac{\lambda_k}{\mu} \right)^{s_k} \right]
\]

Eq. (4.2) shows the second derivative of \( g \).

\[
\frac{d^2 g(S_1, S_2, \ldots, S_n)}{dS_k^2} = \alpha_k = \lambda_k \left[ \frac{\lambda_k}{\mu} \right]^{s_k} \pi_k z_k \log^2 \left( \frac{\lambda_k}{\mu} \right) \frac{h_k z_k \left[ \lambda_k \left( \frac{\lambda_k}{\mu} \right)^{s_k+1} \right] + \pi_k z_k \log^2 \left( \frac{\lambda_k}{\mu} \right) - \lambda_k \mu S_k}{1 - \left( \frac{\lambda_k}{\mu} \right)}
\]

Since \( \frac{\lambda_k}{\mu} < 1 \) and the input parameters (i.e., \( \pi_k, h_k \)) are non-negative, \( \alpha_k, k = 1, 2, 3 \ldots, n \) are greater than zero. In addition the Hessian matrix can be obtained as Eq. (4.3):

\[
H = \begin{bmatrix}
1 & \alpha_1 & 0 & 0 & 0 & 0 \\
2 & 0 & \alpha_2 & \cdots & 0 & 0 \\
\vdots & 0 & \ddots & \ddots & \ddots & \vdots \\
\ell & 0 & 0 & 0 & \alpha_k & \ddots \\
0 & 0 & 0 & \cdots & 0 & \alpha_n \\
0 & 0 & 0 & 0 & 0 & \alpha_n
\end{bmatrix}
\]

Regarding to \( \alpha_k, k=1,2,3 \ldots, n \) are greater than zero, the aforementioned matrix is positive-definite matrix. Accordingly, the objective function is strictly convex. □

**Proposition 2.** For an open DC, the optimal value of the base stock level is equal to \( S_k^* \) or \( S_k^* + 1 \), where, \( \boxed{\cdot} \) operator rounds a real number down to the previous integer. The closed-form for \( S_k^* \) is obtained from Eq. (4.4).
\[
S_k^* = \log \left( \frac{-h_k}{\lambda_k \pi_k \log \left( \frac{\lambda_k}{\mu} \right) - 1} \right) \log \left( \frac{\lambda_k}{\mu} \right) - \frac{h_j \lambda_k}{\mu} \log \left( \frac{\lambda_k}{\mu} \right) - 1
\]

**Proof.** Suppose that \( S_k \) is a continuous variable. Due to the convexity of the objective function (please refer to Proposition 1) the optimal value for the base stock level will be calculated by Eq. (4.5).

\[
\frac{D \text{obj}(S_1, S_2, ..., S_n)}{dS_k} = 0
\]

By replacing \( z_k \) with 1 for open DCs, Eq. (4.4) will be derived. Since the strictly convexity of the objective function (Proposition 1) and being integer the values of the base stock levels for open DCs (see Eq. (3.15)), the best value for the base stock levels can be equal to \( [S_k^*] \) or \( [S_k^*] + 1 \). □

**4.1. The encoding schema and evaluation procedures**

As representation of the solution method for genotype space, the solutions of algorithms are encoded by arrays with the length of \( n \), which \( n \) represents the number of retailers. Each cell of the array represents the corresponding DC that supplies the demands of related retailer. Figure 2 illustrates the encoding schema for the solving procedure. As it can be seen, DC 3 supplies the retailers 1 and 3 and DC 2 supplies the retailer 2.

<table>
<thead>
<tr>
<th>DCs</th>
<th>3</th>
<th>2</th>
<th>3</th>
<th>...</th>
<th>1</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Retailers</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>...</td>
<td>n-1</td>
<td>n</td>
</tr>
</tbody>
</table>

**Figure 2. The encoding schema**

Noteworthy, by determining this array, the location-allocation decisions (i.e., opened DCs and assignments of retailers) are specified. Therefore, we can obtain the pertaining fixed cost of locating DCs and the transportation cost from DCs to the retailers. Nonetheless, in order to obtain total costs for evaluating the solution in the proposed algorithms, it is required that inventory decisions of open DCs are specified too. According to Proposition 2, the optimal inventory policy can be determine for open DCs and consequently we able to obtain the total costs in order to assess the corresponding solution in the proposed algorithms.

**4.2. Simulated annealing**

Simulated annealing is one of the most famous meta-heuristic techniques that its original idea has been inspired from the annealing process of solids and it has been designed to reach a good solution, not necessarily optimum solution within a reasonable computational time. SA has been introduced by Kirkpatrick et al. (1983) as a stochastic algorithm that escapes from local optima by accepting worse solutions with an acceptance probability with regard to freezing rule. The procedure of the algorithm can be implemented by the following steps.

1. Generate an initial solution and then determine the configuration of neighborhood generation.
2. Compare the values of the fitness functions of current solution $E(c)$ and new neighborhood solution $E(n)$. New neighborhood solution $E(n)$ will be accepted with better fitness function value or if $E(n)$ is worse than $E(c)$, the neighbor solution will be accepted due to the Boltzmann probability distribution $e^{-\frac{\Delta E}{T}}$, where $T$ is the temperature of the current iteration.

3. Decrease the temperature $T$ when the equilibrium process is met.

4. Terminate the algorithm when stopping criteria are satisfied.

In this paper, we have used two neighborhood structures. In the first structure, algorithm randomly selects one retailer and one DC (not necessarily to be open in current solution) and then it assigns them to each other. The second structure, the algorithm randomly selects two retailers and then it exchanges their DCs.

The algorithm stops due to different termination criteria such as reaching the minimum temperature, time limitation, no improvement during specific iterations, running algorithm for constant iteration and a set of above conditions. To solve the proposed model, we have considered a set of two conditions including no improvement during specific iterations and reaching the minimum temperature.

The parameters used in SA are defined as following.

\begin{align*}
  NI_{SA-sub} & \quad \text{Maximum number of solutions with no improvement in the sub-loop} \\
  NI_{SA-main} & \quad \text{Maximum number of solutions with no improvement in the main loop} \\
  NI_{sub} & \quad \text{The counter for solutions with no improvement in the sub-loop} \\
  NI_{main} & \quad \text{The counter for solutions with no improvement in the main loop} \\
  SL_0 & \quad \text{Initial solution} \\
  SL & \quad \text{Current solution} \\
  SL' & \quad \text{Solution selected in the neighborhood of } SL \\
  SL_{best} & \quad \text{Best solution} \\
  Obj(SL) & \quad \text{The objective function of solution } SL \\
  CR & \quad \text{Cooling rate} \\
  TM_0 & \quad \text{Initial temperature} \\
  TM_f & \quad \text{Final temperature} \\
  TM & \quad \text{Current temperature}
\end{align*}

The pseudo-code of SA algorithm has been demonstrated in Figure 3.
Input: $T_{M_0}, TM_f, N_{SA-sub}, N_{SA-main}, CR$.
Generate $SL_0$ randomly
$SL_{best} = SL_0, SL = SL_0, TM = TM_f, N_{main} = 0$
While ($N_{main} \leq N_{SA-main}$ and $TM \geq TM_f$) do
  $N_{sub} = 0$
  $BL = False$
  While ($N_{sub} \leq N_{SA-sub}$) do
    Generate $SL'$ in the neighborhood of $SL$
    $\Delta E = Obj(SL') - Obj(SL)$;
    If $\Delta E \leq 0$ Then
      $SL = SL'$,
      If $Obj(SL) < Obj(SL_{best})$ Then
        $SL_{best} = SL$
        $N_{sub} = 0$
        $BL = True$
      Else
        $N_{sub} = N_{sub} + 1$
      End if
    Else
      Generate $ra \rightarrow uniform (0,1)$ randomly
      $\text{If } ra < e^{-\Delta E}$ Then
        $SL = SL'$
      End if
    End if
  End While
  $TM = CR * TM$
  If $BL = True$ then
    $N_{main} = 0$
  Else
    $N_{main} = N_{main} + 1$
  End if
End While

Figure 3. The pseudocode of SA algorithm

4.3. Genetic Algorithm

Similar to SA, GA is a stochastic global search method so that the emergence of this algorithm was investigated in 1960s by Holland (1975). GA works based on the biological evolution behavior and reaches approximate optimal solutions after its main steps including parent selection, recombination, mutation and survivor selection. To select the individuals for next generation two methods can be taken into account. The first method is to eliminate all individuals in older generation and then, to exchange them with new individuals. The second one is to preserve some individuals in old generation with higher values of fitness function and then, to choose other individuals from the new generation.

In our problem, we have generated initial population randomly. DCs and retailers selected randomly will be linked to each other. To select the parents, roulette wheel selection rule (see, Eiben and Smith (2003)) is used. Parents with better values of fitness function are selected to generate offspring.

In order to develop new individuals from existing ones, crossover and mutation operators have been implemented. The crossover operator is used to create new pairs of offspring by combining a pair of parents. In our problem, the tournament selection method and one point crossover is used (see, Eiben and Smith (2003)). Mutation of each individual will be occurred due to the mutation probability $P_m$, and it will change the DC linked to a certain retailer. The probability of the mutation is calculated by Eq. (4.5).
\[ P_m = \frac{1}{\text{number of retailers}} \] (4.5)

Survivor selection or replacement procedure will be performed by Elitism method (see, Eiben and Smith (2003)). Finally, the algorithm will be terminated when no improvement in certain iterations occurs.

The following parameters are used for GA algorithm.

\begin{align*}
\text{NM}or & \quad \text{The number of members in the origin population} \\
\text{NM}ch & \quad \text{The number of members in the children population} \\
\text{NM}mu & \quad \text{The number of members in the mutation population} \\
\text{ST} & \quad \text{Maximum number of solutions with no improvement} \\
\text{NI} & \quad \text{The counter for solutions with no improvement} \\
\text{SL}_{\text{best}} & \quad \text{Best solution} \\
\text{SL}_{\text{ibest}} & \quad \text{Best solution in each iteration} \\
\text{Obj(} SL \text{)} & \quad \text{The objective function of the solution} \text{SL} \\
\text{Pr}_{\text{Cr}} & \quad \text{The probability of crossover} \\
\text{Pr}_{\text{Mu}} & \quad \text{The probability of mutation}
\end{align*}

In this paper, three populations including origin, children and mutation are considered, where the numbers of their members are \( N_{\text{pop}} \), \( N_{\text{ch}} \) and \( N_{\text{mu}} \), respectively. Parameters \( N_{\text{ch}} \) and \( N_{\text{mu}} \) are equal to \( 2 \times \left\lfloor \frac{N_{\text{pop}} \times \text{Pr}_{\text{Cr}}}{2} \right\rfloor \) and \( \left\lfloor N_{\text{pop}} \times \text{Pr}_{\text{Mu}} \right\rfloor \) respectively, where \( \left\lfloor \cdot \right\rfloor \) rounds a real number down to the previous integer.

The pseudo-code of GA algorithm is illustrated in Figure 4.

---

**Input:** \( \text{NM}or, \text{Pr}_{\text{Cr}}, \text{Pr}_{\text{Mu}}, \text{ST} \).

Generate \( \text{NM}or \) solutions randomly to build the origin population.

Evaluate candidate solutions of the origin population.

Find the best solution in the origin population. Name it \( \text{SL}_{\text{best}} \).

\( \text{NI} = 0 \)

\[ \text{While (NI \leq ST) do} \]

\( I = 1 \)

\[ \text{While (I \leq \frac{\text{NM}ch}{2}) do} \]

Select a pair of parents from the origin population and recombine them

recombine the selected parents and create offsprings

Insert the created offsprings in the children population

\( I = I + 1 \)

End while

\( I = 1 \)

\[ \text{While (I \leq \text{NM}mu) do} \]

Select a parent from the origin population
Modify the selected parent

Insert the modified parent in the mutation population

\[ I = I + 1 \]

**End while**

Merge the origin, children and mutation populations and generate a new population.

Evaluate solutions of the generated population.
Find the best solution in the generated population. Name it \( SL_{ibe} \)

\[ \text{If } Obj(SL_{ibe}) < Obj(SL_{Best}) \text{ Then} \]

\[ SL_{Best} = SL_{ibe} \]

\[ NI = 0 \]

**Else**

\[ NI = NI + 1 \]

**End if**

**End While**

*Figure 4. The Pseudocode of GA algorithm*

### 4.4. Parameter setting

Parameter setting has significant impacts on the performance of the meta-heuristic algorithms. Therefore, we implement Taguchi method in order to tune the parameters of the proposed algorithms. Taguchi is a powerful method for robust parameter design which is based on reducing the effect of causes of variation and improving the quality of solutions. This method provides analysis of different parameters with no need of high amount of experiments. Taguchi takes advantages from two significant tools, the signal-to-noise ratio (S/NR) and orthogonal array (OA). S/NR is a measure that intends to mix information about the variance and the mean. Variety of response, i.e. the nominal or target value, is measured by S/NR under different noise conditions. By using S/NR, control factors that diminish variability are determined and then those control factors that move the mean to target with small or no effect on the S/NR will be specified. In Taguchi method, OA is implemented in order to evaluate the impact of many parameters on the performance of the algorithm in a compressed set of experiments. The suitable OA can be chosen by determining the number of parameters and the number of levels. Accordingly, first the concerning parameters are specified and then their levels of varying will be determined. The levels of a parameter usually consisting of the minimum, maximum, and current values can be determined by an in-depth knowing of the process. For each parameter, when the difference between the maximum and minimum is large, more levels need to test and as the range of a parameter is small, fewer values can be tested. For more detailed insights, interested readers can refer to Taguchi et al. (2000).

In this research, four factors including \( TM_0, CR, NI_{SA-sub} \) and \( NI_{SA-main} \) are taken into account for SA and each factor are studied in three levels. Table 2 shows the levels for SA factors. Additionally, three factors including \( NMor, Pr_{Cr} \) and \( ST \) are considered for GA that the concerning levels are given in Table 3. Using MINITAB 16, we have first created Taguchi designs for the solving algorithms and then the experiments are run based on the created designs. For both algorithms, \( L_9 \) OA have been chosen. Design factors for SA and GA are displayed in Tables 4 and 5, respectively. After analyzing the performed experiments by Taguchi method, the appropriate level of each factor can be obtained.
Here, that level that increases the S/NR measure is selected. Figures 5 and 6 show the average S/NR for considered factors of algorithms. It can be observed that the best values for SA factors, i.e. $TM_0$, $CR$, $NIL_{SA-sub}$ and $NIL_{SA-main}$, are 950, 0.95, 150 and 300, respectively. Besides, for GA the best values of $NMor$, $Pr_Cr$ and $ST$ are 140, 0.8 and 1000, respectively.

### Table 2. Levels of SA factors

<table>
<thead>
<tr>
<th>Levels</th>
<th>$TM_0$</th>
<th>$CR$</th>
<th>$NIL_{SA-sub}$</th>
<th>$NIL_{SA-main}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Level 1 (low)</td>
<td>850</td>
<td>0.9</td>
<td>50</td>
<td>100</td>
</tr>
<tr>
<td>Level 2 (medium)</td>
<td>950</td>
<td>0.95</td>
<td>100</td>
<td>200</td>
</tr>
<tr>
<td>Level 3 (large)</td>
<td>1050</td>
<td>0.99</td>
<td>150</td>
<td>300</td>
</tr>
</tbody>
</table>

### Table 3. Levels of GA factors

<table>
<thead>
<tr>
<th>Levels</th>
<th>$NMor$</th>
<th>$Pr_Cr$</th>
<th>$ST$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Level 1 (low)</td>
<td>100</td>
<td>0.6</td>
<td>300</td>
</tr>
<tr>
<td>Level 2 (medium)</td>
<td>120</td>
<td>0.7</td>
<td>750</td>
</tr>
<tr>
<td>Level 3 (large)</td>
<td>140</td>
<td>0.8</td>
<td>1000</td>
</tr>
</tbody>
</table>

### Table 4. Taguchi designs for SA

<table>
<thead>
<tr>
<th>Design</th>
<th>$TM_0$</th>
<th>$CR$</th>
<th>$NIL_{SA-sub}$</th>
<th>$NIL_{SA-main}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>850</td>
<td>0.9</td>
<td>50</td>
<td>100</td>
</tr>
<tr>
<td>2</td>
<td>850</td>
<td>0.95</td>
<td>100</td>
<td>200</td>
</tr>
<tr>
<td>3</td>
<td>850</td>
<td>0.99</td>
<td>150</td>
<td>300</td>
</tr>
<tr>
<td>4</td>
<td>950</td>
<td>0.9</td>
<td>100</td>
<td>300</td>
</tr>
<tr>
<td>5</td>
<td>950</td>
<td>0.95</td>
<td>150</td>
<td>100</td>
</tr>
<tr>
<td>6</td>
<td>950</td>
<td>0.99</td>
<td>50</td>
<td>200</td>
</tr>
<tr>
<td>7</td>
<td>1050</td>
<td>0.9</td>
<td>150</td>
<td>200</td>
</tr>
<tr>
<td>8</td>
<td>1050</td>
<td>0.95</td>
<td>50</td>
<td>300</td>
</tr>
<tr>
<td>9</td>
<td>1050</td>
<td>0.99</td>
<td>100</td>
<td>100</td>
</tr>
</tbody>
</table>

### Table 5. Taguchi designs for GA

<table>
<thead>
<tr>
<th>Design</th>
<th>$NMor$</th>
<th>$Pr_Cr$</th>
<th>$ST$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>100</td>
<td>0.6</td>
<td>800</td>
</tr>
<tr>
<td>2</td>
<td>100</td>
<td>0.7</td>
<td>900</td>
</tr>
<tr>
<td>3</td>
<td>100</td>
<td>0.8</td>
<td>1000</td>
</tr>
<tr>
<td>4</td>
<td>120</td>
<td>0.6</td>
<td>900</td>
</tr>
<tr>
<td>5</td>
<td>120</td>
<td>0.7</td>
<td>1000</td>
</tr>
<tr>
<td>6</td>
<td>120</td>
<td>0.8</td>
<td>800</td>
</tr>
<tr>
<td>7</td>
<td>140</td>
<td>0.6</td>
<td>1000</td>
</tr>
<tr>
<td>8</td>
<td>140</td>
<td>0.7</td>
<td>800</td>
</tr>
<tr>
<td>9</td>
<td>140</td>
<td>0.8</td>
<td>900</td>
</tr>
</tbody>
</table>
Figure 5. S/NR ratio plot for SA factors
5. Computational Results

In this section, the performance of proposed model and meta-heuristic algorithms are evaluated in terms of the quality of the solutions and computation time for the numerical experiments. Therefore, both solving algorithms are coded in Java programming language and they are run on an Intel(R) (TM) 2.50Hz CPU with 4G memory. The parameters of the model have been generated randomly and they are illustrated in Table 6. We generated three sizes test problems consisting of small, medium and large sizes. In small instances, the performance of the solving algorithms has been compared with General Algebraic Modeling System (GAMS) – version 24.1 - where the BARON solver has been implemented. The obtained results such as objective function, CPU Time and gaps between the algorithms and GAMS are reported in Table 7. The gap is calculated using Eq. (5.1).

\[
\text{Gap(\%)} = \frac{\text{obj}_{GS} - \text{obj}_{PA}}{\text{obj}_{GS}} \times 100
\]  

(5.1)

Where, \( \text{obj}_{GS} \) and \( \text{obj}_{PA} \) are the value of objective functions obtained by GAMS and the related meta-heuristic algorithm, respectively. In other words, Gap1 and Gap2 are the representatives of GA and SA, respectively. For instances 1 and 2, we have observed that SA and GA are able to obtain same solutions with those obtained by GAMS within a reasonable time. Likewise, our algorithms find the better solutions for instances 4 and 5 with considerably less time than the BARON’s run time. Thus, it can be concluded that the solving algorithms perform better than GAMS solver. The algorithms have been run for medium and large instances and the results including the values of objective function, CPU time and the gap are shown in Table 8. The gap of solving algorithms is calculated according to Eq. (5.2).

\[
\text{Gap(\%)} = \frac{\text{obj}_{GA} - \text{obj}_{SA}}{\text{obj}_{GA}} \times 100
\]  

(5.2)

Where, \( \text{obj}_{GA} \) and \( \text{obj}_{SA} \) are the value of objective functions for GA and SA, respectively. As the numerical example shows, the quality of solutions of GA is better than those ones obtained by SA, where the maximum gap is 0.853%. On the other hand, in term of elapsed times, SA performs better. Furthermore, elapsed times for both proposed algorithms are increased by increasing the sizes of instances, as expected.
Thus, the number of open DCs can provide the ordered products for open retailers. This fact is consistent to the findings of Sadjadi et al. (2016). Table 10 shows the impacts of the demand rates on the number of open DCs and the value of the objective function.

### 5.1. Sensitivity analysis

In this section, the impact of parameters on the number of open DCs, the objective function and the base stock level is investigated.

#### 5.1.1. The number of open DCs

The effects of the value changing of \( \mu \) on the number of open DCs and the objective function for two data sets, 10×25 and 21×80 (\( n \times m \) means that the test problems include \( n \) potential DCs and \( m \) retailers), are investigated and the results are reported in Table 9. It can be inferred from Table 9 that the more the value of \( \mu \) increases, the more the number of open DCs decreases. Because by increasing the value of \( \mu \), the expected lead-time decreases and consequently the manufacturers can provide the ordered products for open DCs in a shorter time. Thus, the shortages of each open DC decreases and more demands can be met. Therefore, the SC tends to reduce the number of open DCs in order to decrease the annual fixed cost of locating DCs. Also, it can be observed that the more the value of \( \mu \) increases, the more the objective function decreases. This fact is consistent to the findings of Sadjadi et al. (2016). Table 10 shows the impacts of the demand rates on the number of open DCs and the value of the objective function.

### Table 6. The values of the parameters.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>( c_k )</th>
<th>( A_k )</th>
<th>( h_k )</th>
<th>( \pi_k )</th>
<th>( F_k )</th>
<th>( \lambda_i )</th>
<th>( T_{ki} )</th>
</tr>
</thead>
</table>

### Table 7. Computational results for small instances.

<table>
<thead>
<tr>
<th>NO.</th>
<th># Retailer</th>
<th>Potential DCs</th>
<th>GAMS</th>
<th>GA</th>
<th>SA</th>
<th>Gap GA</th>
<th>Gap SA</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>Cost ($)</td>
<td>CPU Time (Sec)</td>
<td>Cost ($)</td>
<td>CPU Time (Sec)</td>
<td>Cost ($)</td>
</tr>
<tr>
<td>1</td>
<td>4</td>
<td>2</td>
<td>182644</td>
<td>1041</td>
<td>182644</td>
<td>0.037</td>
<td>182644</td>
</tr>
<tr>
<td>2</td>
<td>10</td>
<td>4</td>
<td>476768</td>
<td>3461</td>
<td>476768</td>
<td>0.094</td>
<td>476768</td>
</tr>
<tr>
<td>3</td>
<td>15</td>
<td>6</td>
<td>672898</td>
<td>8751</td>
<td>672652</td>
<td>0.152</td>
<td>672659</td>
</tr>
<tr>
<td>4</td>
<td>20</td>
<td>8</td>
<td>859999</td>
<td>11432</td>
<td>858623</td>
<td>0.179</td>
<td>858651</td>
</tr>
</tbody>
</table>

### Table 8. Computational results for medium and large instances.

<table>
<thead>
<tr>
<th>NO.</th>
<th># Retailer</th>
<th>Potential DCs</th>
<th>GA</th>
<th>SA</th>
<th>Gap GA</th>
<th>Gap SA</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>Cost ($)</td>
<td>CPU Time (Sec)</td>
<td>Cost ($)</td>
<td>CPU Time (Sec)</td>
</tr>
<tr>
<td>5</td>
<td>25</td>
<td>10</td>
<td>1069945</td>
<td>1.176</td>
<td>0.345</td>
<td>1069945</td>
</tr>
<tr>
<td>6</td>
<td>35</td>
<td>12</td>
<td>1529056</td>
<td>1.323</td>
<td>0.384</td>
<td>1530294</td>
</tr>
<tr>
<td>7</td>
<td>45</td>
<td>15</td>
<td>1910099</td>
<td>1.528</td>
<td>0.551</td>
<td>1912923</td>
</tr>
<tr>
<td>8</td>
<td>55</td>
<td>17</td>
<td>2246673</td>
<td>2.562</td>
<td>0.592</td>
<td>2249941</td>
</tr>
<tr>
<td>9</td>
<td>70</td>
<td>19</td>
<td>2932897</td>
<td>3.443</td>
<td>0.660</td>
<td>2949496</td>
</tr>
<tr>
<td>10</td>
<td>80</td>
<td>21</td>
<td>3194120</td>
<td>13.37</td>
<td>1.224</td>
<td>3197255</td>
</tr>
<tr>
<td>11</td>
<td>95</td>
<td>28</td>
<td>3671902</td>
<td>21.96</td>
<td>1.551</td>
<td>3684850</td>
</tr>
<tr>
<td>12</td>
<td>100</td>
<td>35</td>
<td>4067175</td>
<td>25.99</td>
<td>1.990</td>
<td>4073398</td>
</tr>
<tr>
<td>13</td>
<td>110</td>
<td>40</td>
<td>4255463</td>
<td>36.625</td>
<td>2.045</td>
<td>4258169</td>
</tr>
<tr>
<td>14</td>
<td>140</td>
<td>45</td>
<td>5548268</td>
<td>51.12</td>
<td>2.511</td>
<td>5586842</td>
</tr>
<tr>
<td>15</td>
<td>150</td>
<td>50</td>
<td>5833398</td>
<td>85.43</td>
<td>3.192</td>
<td>5852301</td>
</tr>
</tbody>
</table>

Journal of Industrial Engineering and Management Studies (JIEMS), Vol.6, No.2
It can be concluded from Table 10 that the more the demand rates of retailers increases, the more open DCs to meet the demands needs in order to decrease the shortage cost. Furthermore, increasing the values of demand rates enlarges the total costs of SC including the fixed cost of locating DCs, the transportation cost from DCs to retailers and the annual inventory costs.

Table 9. The effects of the value changing of $\mu$ on the number of open DCs and the objective function.

<table>
<thead>
<tr>
<th>The value of $\mu$</th>
<th># open DCs</th>
<th>Cost($)</th>
<th># open DCs</th>
<th>Cost($)</th>
</tr>
</thead>
<tbody>
<tr>
<td>2000</td>
<td>9</td>
<td>1135738</td>
<td>21</td>
<td>3678297</td>
</tr>
<tr>
<td>3000</td>
<td>7</td>
<td>1089615</td>
<td>18</td>
<td>3470915</td>
</tr>
<tr>
<td>4000</td>
<td>6</td>
<td>1079730</td>
<td>16</td>
<td>3322724</td>
</tr>
<tr>
<td>5000</td>
<td>6</td>
<td>1069991</td>
<td>16</td>
<td>3196149</td>
</tr>
<tr>
<td>6000</td>
<td>6</td>
<td>1065491</td>
<td>16</td>
<td>3084758</td>
</tr>
</tbody>
</table>

Table 10. The effects of demand rates on the number of open DCs and the objective function.

<table>
<thead>
<tr>
<th>The mean of annual demand</th>
<th># open DCs</th>
<th>Cost($)</th>
<th># open DCs</th>
<th>Cost($)</th>
</tr>
</thead>
<tbody>
<tr>
<td>2000</td>
<td>4</td>
<td>701792.8</td>
<td>5</td>
<td>1214401</td>
</tr>
<tr>
<td>3000</td>
<td>6</td>
<td>1049009</td>
<td>9</td>
<td>2474990</td>
</tr>
<tr>
<td>4000</td>
<td>6</td>
<td>1396694</td>
<td>13</td>
<td>3777726</td>
</tr>
<tr>
<td>5000</td>
<td>7</td>
<td>1748224</td>
<td>15</td>
<td>5101892</td>
</tr>
<tr>
<td>6000</td>
<td>8</td>
<td>2107352</td>
<td>21</td>
<td>6480549</td>
</tr>
</tbody>
</table>

5.1.2. Base stock

In his research, we have obtained the optimal quantity of base stock level. Here, we have surveyed the impacts of service rate and demand rate on the base stock level. For this sake, we have assumed $h=30, \pi =75, \lambda =445$ and $\mu=610$. The relationship between the demand rate and base stock level is pictured by Figure 8. It can be concluded from Figure 8 that the more the demand rates of retailers increase, the more the base stock level enlarges. Its reason is due to increasing the probability of shortage and the value of demand rate and consequently the base stock level will enlarge in order to reduce the impact of confronting with shortages.

![Figure 8. Sensitivity analysis of the base stock level with respect to changes of demand rate](image)
6. Conclusion and Future Research

In this paper, a multi-echelon stochastic SC network design problem in order to minimize the total expected location, inventory and transportation costs was investigated. To bring more reality to the proposed model, the demand and lead-time were assumed to be hemmed in by uncertainty. In an attempt to tackle the proposed supply chain network design problem, a two-phase approach based on queuing and optimization models was devised. Exploiting the queuing approach, the first phase captured the inherent uncertainty of parameters. In the second phase, a mixed-integer nonlinear model was employed to formulate the proposed supply chain network design problem. As such, in the light of the concerned problem was NP-hard, two meta-heuristic algorithms were devised to solve the instances in small, medium and large sizes within a reasonable time. Results endorsed high efficiency and effectiveness of the solving algorithms in comparison with GAMS solver. Meanwhile, we proved the convexity of the objective function and achieved the optimal value for the base stock level as a closed-form. Various experiments were conducted through which appealing insights were reached. For example, we showed the solving algorithms can solve the proposed problem in efficient ways. An another interesting result was that as the value of $\mu$ increases, the expected lead-time decreases and consequently the manufacturers can provide the ordered products for open DCs in a shorter time. As such, with increasing the demand rate of retailers, more open DCs are needed in order to decrease the shortage costs.

For future research, different areas to develop can be accounted as follows. 1) Developing the problem as a multi-product and multi-period model. 2) Utilizing the exact methods to solve the problem. 3) Taking into account multiple inventory policies in DCs and 4) Bringing the concepts of reliability into the model.

References


Queuing approach and optimal inventory decisions in a stochastic supply chain network design


