A Benders' Decomposition Method to Solve Stochastic Distribution Network Design Problem with Two Echelons and Inter-Depot Transportation

Vahid Reza Ghezavati*
Faculty of Industrial Engineering, Islamic Azad University, South Tehran Branch, Tehran, Iran
Received 15 September, 2014; Revised 05 March, 2015; Accepted 24 June, 2015

Abstract
In many practical distribution networks, managers face significant uncertainties in demand, local price of building facilities, transportation cost, and macro and microeconomic parameters. This paper addresses design of distribution networks in a supply chain system which optimizes the performance of distribution networks subject to required service level. This service level, which is considered for each arbitrary request arriving at a distribution center (facility), has a (pre-specified) small probability of being lost. In this mathematical model, customer’s demand is stochastic that follows uniform distribution. In this model, inter-depot transportation (transportation between distributions centers (DCs)), capacities of facilities, and coverage radius restrictions are considered. For this restriction, each DC cannot service all customers. The aim of this model is to select and optimize location of plants and DCs. Also, the best flow of products between DCs and from plants to DCs and from DCs to customers will be determined. The paper presents a mixed integer programming model and proposed an exact solution procedure in regard to Benders’ decomposition method.

Keywords: Facility location, Distribution network, Benders’ Decomposition, Coverage Radius, Uncertainty modeling, Inter-depot transportation.

1. Introduction

It is quite common nowadays to see manufacturers and retailers joining efforts to efficiently handle the flow of products and to closely coordinate the production and supply chain system. An important strategic aspect related to the design and operation of a physical distribution network in a supply chain system is to determine the best sites for intermediate warehouses, or distribution centers. The use of DCs provides a company with flexibility to respond to demand fluctuation in the marketplace and can result in significant cost savings due to economies of scale in transportation or shipping costs.

Many researchers have extensively studied facility location and demand allocation problems. Previous research studies are well reviewed by Francis, McGinnis, and White (1983), Aikens (1985), Brandeau and Chiu (1989), and Avella et al. (1998). More recently, Jayaraman (1998) considered the capacitated warehouse location problem that involves locating a given number of warehouses to satisfy customer demands for different products. Pirkul and Jayaraman (1998) developed the previous problem by allowing locating also a given number of plants. They formulated the problem as a mixed integer model and developed a Lagrangean-based heuristic solution procedure. Tragantaleignsak, Holt, and Ronnqvist (2000) considered a bi-echelon facility location problem in which the facilities in the first echelon are assumed to be uncapacitated but the facilities in the second echelon are capacitated. The aim of their effort is to determine the number and locations of facilities in both echelons in order to satisfy customer demand per each product. They proposed a Lagrangean relaxation based branch and bound algorithm to solve the problem. Gourdin, Labbe, and Laporte (2000) studied a different type of the uncapacitated facility location problem where two customers allocated to the same facility are matched. They developed several methods to solve the problem after obtaining valid inequalities, and optimality cuts for the problem.

Current, Daskin, and Schilling (2001) and Daskin and Owen (1999) overview both deterministic and stochastic facility location. For a further detailed study of facility location theory, see the contents by Daskin (2013), Drezer (1995), or Hurter and Martinich (1989). Sheppard (1974) was one of the first authors proposed a mixed integer model and developed a Lagrangean-based heuristic solution procedure.

* Corresponding author Email address: V.ghezavati@azad.ac.ir
scenario approach to facility location. He suggests selecting facility locations to minimize expected cost; however, he does not discuss the subject at length.

In any stochastic programming problem, one must determine which decision variables are first-stage and which are second-stage; that is, which variables must be set now and which may be set after the uncertainty has been resolved. In stochastic location modeling, locations are generally first-stage decisions while assignments of customers to facilities are second-stage decisions (Ghezvati, Jabal-Ameli, & Makui (2008) and Snyder, 2006). If both decisions occur in the first stage, most stochastic location problems can be reduced easily to deterministic problems.

Mirchandani, Oudjit, and Wong (1985) and Weaver and Church (1983) presented algorithms for a multi-scenario for P-median problem (PMP). Their algorithms effectively treat the problem as a deterministic PMP with [I\times|S|] customers instead of |I|, where I is the set of customers and S is the set of scenarios. Louveaux (1986) introduced stochastic versions of the capacitated PMP and Capacitated facility location problem (CFLP) in which demand, production costs, and selling prices are arbitrary. Vidal and Goetschalckx (2000) discussed the importance of incorporating various types of uncertainty into global supply chain design decisions.

More formal stochastic programming techniques are used by Alonso-Ayuso, Escudero, García, Ortúñan, and Pérez (2003), to solve multi-echelon supply chain design problems.


Moin, Salhi, and Aziz (2011) addressed an inventory routing, many-to-many distribution network consisting of an assembly plant and many distinct suppliers where each supplies a distinct product. Hiassat and Diabat (2011) studied the LRI problem with perishable product, through a multi-period model.

In this paper, we define a notation of a supply chain system considering stochastic demand and inert-depot transportation. Also some restrictions such as coverage radius and inventory capacity are assumed in the thinking process.

The structure of this paper is as follows. In Section 2, we present the formulation of the problem. Section 3 proposes solution procedure regarded to Benders’ decomposition method. We present computational results and sensitivity analysis in Section 4. In Section 5, we summarize our conclusions and discuss avenues for future research.

2. Model Formulation

Let N be the set of customers, which face uniform distributed demands that are independent among customers. Let M be the set of potential sites for distribution centers. The firm pays a fixed location cost for opening a DC, as well as a holding cost for inventory. In this model since customers’ demand is stochastic, service level constraint is considered for that an arbitrary request arriving at a DC, should only have a (pre-specified) small probability of being lost.

The notice is that all distribution centers cannot service all of the customers due to considering special coverage restriction for each DC. Thus, if a customer cannot fall in the coverage radius, so the DC cannot service that customer. In this model, this point is considered which makes the model to be more realistic.

The other assumption applied in this study is inter-depot transportation. This point can enable DCs to have transportation products among each other. Due to stochastic demand which leads to inventory fluctuation in facilities, inter – depot transportation has risk-pooling effect on the supply chain network. Thus, holding inventory and shortage in the whole chain will be reduced. The sample of representation of a distribution network considering inter – depot transportation is illustrated in Figure 1.

In the previous location - inventory models the required order, for each DC’s, is the sum of two parts. First part is the sum of mean demand, assigned to the DC and the second part is the safety stock. A new approach for this model in this article can be proposed without dividing the order to two parts. By combining the two parts, let $M_i$ be the amount of the order DC $i$. In this section, we aim to develop a chance constraint programming in order to prevent demand shortage for each customer that is assigned to a DC. As it was defined earlier, the demand for each customer follows uniform distribution within the interval $(a_i, b_i)$. For this purpose, a amount of product that a DC orders to cover each customer's demand $(Q_i)$ must satisfy the following constraint. By this way, the probability that each customer faces to demand shortage is at most $\alpha\%$. 

```text
28
```
A new constraint must be added for this purpose and thus, we don’t need to compute the safety stock.

\[ P \left( \text{an arbitrary request arriving by customer } i \text{ at DC } j \text{ with inventory level } Q_{ij} \text{ is not lost} \right) \geq 1 - \alpha \]  

(1)

By computing above constraint, the following constraint will be considered in the mathematical model. According to this, order quantity is greater than customers’ mean demand and model determines optimum quantity so that customers’ demand satisfies at least \((1-\alpha)\)% , so there is no necessity to hold safety stock. The constraint considered for this is

\[ Q_{ij} \geq (1 - \alpha) \times (b_j - a_i) + a_i \]

(2)

Amount of product that is found by above chance constraint is valid, once customer \(i\) is assigned to DC \(j\). so, this constraint is modified as follows:

\[ Q_{ij} \geq [(1-\alpha)(b_j - a_i) + a_i] \times X_{ij} \]

(3)

Also, let \(\eta_{ij}\) be the probability that customer \(i\) which is assigned to DC \(j\) faces to product shortage. So, amount of \(\eta_{ij}\) can be computed as follows:

\[ \Pr(x_i \leq Q_{ij}) + \eta_{ij} = 1 \Rightarrow \eta_{ij} = \frac{1 \times (b_j - a_i) - Q_{ij} + a_i}{b_j - a_i} \]

(4)

Note: Above equation leads to risk pooling of the system since the demand of customers follows probabilistic structure and some customers request demand less than decision maker’s estimation and some of them request demand more than decision maker’s estimation.
2.1 Parameters and Decision Variables

Parameters
- \( N \) index set of customer zones.
- \( M \) index set of potential distribution sites.
- \( L \) index set of potential plant sites.
- \( C_{ij} \) Cost of supplying one unit of demand to customer zone \( i \) from \( DC \) at site \( j \).
- \( \overline{C}_{jk} \) Cost of supplying one unit of demand to \( DC \) at site \( j \) from plant \( k \).
- \( F_j \) Fixed cost for opening \( DC \) at site \( j \).
- \( G_k \) Fixed cost for opening plant at site \( k \).
- \( b_j \) Capacity for the potential \( DC \) at site \( j \).
- \( e_k \) Capacity for the potential plant at site \( k \).
- \( U_i \sim (a_i, b_i) \) Uniform distribution function for demand of customer \( i \).
- \( \lambda_i \) Mean of demand per unit time for customer \( i \). (\( \lambda_i = \frac{a_i + b_i}{2} \))
- \( h_i \) Holding cost per unit in \( DC \) at site \( j \).

\[
Z_{ij} = \begin{cases} 
1 & \text{if the } DC \text{ at } j \text{ covers the customer site } i. \\
0 & \text{Otherwise}
\end{cases}
\]

Decision variables:
- \( X_{ij} \) 1 if demand of customer \( i \) satisfied by \( DC \) at site \( j \).
- \( Q_{jk} \): Amount of product that \( j \) orders to satisfy demand of customer \( i \) with the probability at least 1 - \( \alpha \).
- \( Y_{jk} \) Amount of product that is transported from plant \( k \) to \( DC \).
- \( U_j = \begin{cases} 
1 & \text{if a } DC \text{ is located at site } j. \\
0 & \text{Otherwise}
\end{cases} \)
- \( V_k = \begin{cases} 
1 & \text{if a plant is located at site } k. \\
0 & \text{Otherwise}
\end{cases} \)
- \( IDT_{j,j'} \) Total product shipped from \( DC \) to \( DC' \).

2.2 Proposed Model

\[
\min Z = \sum_{j \in M} \sum_{k \in L} C_{jk} \times Y_{jk} + \sum_{j \in M} \sum_{i \in N} \lambda_i \times X_{ij} \times C_{ij} + \sum_{j \in M} F_j \times U_j + \sum_{k \in L} G_k \times V_k + \sum_{i \in N} \sum_{j \in M} \lambda_i \times X_{ij} \times LI \times \eta_{ij}
\]

St:
- \( \sum_{j \in M} X_{ij} = 1 \quad \forall i \in N \) (6)
- \( X_{ij} \leq Z_{ij} \times U_j \quad \forall i \in N, \forall j \in M \) (7)
- \( \sum_{k \in L} Y_{jk} \leq U_j \times b_j \quad \forall j \in M \) (8)
\[ Q_{ij} \geq [(1 - \alpha)(b_j - a_i) + a_i]X_{ij} \]  
\[ MJ = \sum_i Q_{ij} \]  
\[ \eta_{ij} = \frac{1 \times (b_j - a_i) - Q_{ij} + a_i}{b_j - a_i} \]  
\[ MJ = \sum_{k,M} Y_{jk} + \sum_{j,M,j' \neq j} IDT_{f,j} - \sum_{j,M,j' \neq j} IDT_{j,f} \] \\forall j \in M  
\[ \sum_{f,M,j' \neq j} IDT_{j,f} \leq \sum_{k,M} Y_{jk} \] \\forall j \in M  
\[ \sum_{i \in N} X_{ij} \leq b_j \times U_j \] \\forall j \in M  
\[ \sum_{j \in M} Y_{jk} \leq \epsilon_k \times V_k \] \\forall j \in M  
\[ X_{ij}, U_j, V_k \in \{0,1\} \]  
\[ Y_{jk}, IDT_{j,f} \geq MJ, Q_{ij} \geq 0 \]  

This model minimizes total expected cost made of: the costs to serve the demand of customers from the DCs, the costs of shipments from the plants to the DCs, the costs associated with opening and operating the DCs and plants, the expected costs of inventory which is lost in any opened DC (the expected cost = probability of being lost for demand \times total expected demand), and total costs of ordering and holding inventory. Constraint set (6) ensures that all customers must be allocated to DCs. Constraint set (7) says that a customer can be allocated to a DC once the customer is in the coverage radius of DC and also, DC is opened. Constraint set (8) guarantees that the total product shipped from plants to DCs should be less than DC’s capacity. Set constraint (9) ensures that service level for DCs would be at least (1-\alpha) %. Set constraints (10) computes total products that each DC must order to cover all assigned customers subject to service level limitation. Constraint set (11) present total net inventory that DC has and can satisfies customers’ demand and ships to the other DCs by that. Set constraint (12) computes the probability that each customer faces to product shortage. Constraint set (13) says that total product shipped from a DC to the customers should be less than total product received from plants and others depots. Constraint set (14) specifies that total averagedemand that a DC satisfies should be less than capacity of the DC. Constraint set (15) shows capacity constraint for plants. Constraints set (16), (17) determine type of the variables.

2.3 Linearization of the proposed model

The last term in the objective function is appeared nonlinear because a binary variable \((X_{ij})\) is multiplied to a continuous variable \((\eta_{ij})\). This term can be converted to a linear one by the algorithm that is introduced by Ghezavati and Saidi-Mehrabad (2011). To apply this method the following auxiliary constraints are added to the model.

\[ LP_{ij} = X_{ij} \times \eta_{ij} \]  
\[ LP_{ij} \leq M \times X_{ij} \]  
\[ LP_{ij} \leq \eta_{ij} \]  
\[ LP_{ij} \geq \eta_{ij} - M(1 - X_{ij}) \]

3. Solution Procedure

All the decision variables except location variables are continuous for proposed model. The model without binary variables \((W_j)\) is a linear programming solved easily. Experimental studies of our model, obtained by LINGO and CPLEX solver, demonstrate the running times to very long (more than 3 hours) for the instances that have large number (more than 30) of the binary variables. Also, Geoffrion and Powers (1995) show distribution systems design become extremely difficult for the associated large-scale models to solve optimality without the implementation of Benders’ decomposition or factorization methods. For this characteristic, Benders (1962) presents a solution procedure called Benders’ Decomposition algorithm. In this way, the problem decomposes into two distinct problems called as sub-problem and master problem. The master problem is established by part of original objective function that contains complicated variables as its objective function, and constraints of original problem includes only the complicated variables. The reminder of original problem makes sub-problem which its complicated variables are fixed by solution of master problem. Objective functions of the master problem and the sub-problem obtain respectively upper bound and lower bound of the original objective functions to maximize problem. The algorithm tries to close the gap between these bounds through using optimality cuts in the master problem. The optimality cuts are formulated by dual values of complicated constraints.
in sub-problem. If sub-problem becomes infeasible, feasibility cut will be produced in the master problem. We adopt a solution procedure as Benders’ Decomposition algorithm. In the proposed algorithm, master problem and sub-problem are formulated in regard to following indices and parameters.

Indices of algorithm:

\( \text{iter} \in I \) \quad \text{Iterations of the algorithm}
\( p \subseteq I \) \quad \text{Set of iterations that sub-problem is feasible}
\( f \subseteq I \) \quad \text{Set of iterations that sub-problem is infeasible}

Parameters of algorithm:

\( W_j \) \quad \text{Set of binary variables}
\( \text{fix}_W\_{j(\text{iter})} \) \quad \text{Fixed value for } W_j \text{ at iteration iter}
\( u_{j(\text{iter})} \) \quad \text{Dual value in relation to constraints (8) and (9) at iteration iter}
\( \text{obj}_{\text{sub}}\_{\text{iter}} \) \quad \text{Value of objective function for sub-problem at iteration iter}
\( \text{LB} \) \quad \text{Lower bound of problem}
\( \text{UB} \) \quad \text{Upper bound of problem}

Sub-problem:

\[
\begin{align*}
\text{Max} & \quad u_1 \times \left( \sum_{k \in L} Y_{jk} - \text{fix} \_ U_j \times b_j \right) + \\
& \quad u_2 \times \left( (1 - \alpha) \times (b_j - a_j) + a_j \right) \times \text{fix} \_ X_{\text{fix}} - Q_j \\
\text{Subject to all constraints expect those ones contain binary variables.} & \quad \sum_{k \in L} Y_{jk} \leq \text{fix} \_ U_j \times b_j \quad \forall j \in M \\
& \quad Q_{i,j} \geq [(1 - \alpha) \times (b_j - a_j) + a_j] \times \text{fix} \_ X_{\text{fix}} \\
\end{align*}
\]

Master problem:

\[
\begin{align*}
\lambda & \quad = Z \quad \text{where binary variables are fixed} + \\
& \quad u_1 \times \left( \sum_{k \in L} Y_{jk} - \text{fix} \_ U_j \times b_j \right) + \\
& \quad u_2 \times \left( (1 - \alpha) \times (b_j - a_j) + a_j \right) \times \text{fix} \_ X_{\text{fix}} - Q_j \\
& \quad \sum_{j : \text{fix} \_ W_j = 0} W_j + \sum_{j : \text{fix} \_ W_j = 1} (1 - W_j) \geq 1 \quad \forall f
\end{align*}
\]

In the objective function of master problem, \( \alpha \) is a free variable that exists in optimality cuts. At first, master problem without \( \alpha \) and cuts solves then optimal values are fixed in sub-problem. In turn, if the sub-problem is feasible, optimality cut will be placed in master problem or if it is infeasible, feasibility cut will be located in master problem. Consequently, master problem obtained by optimality cuts or feasibility cuts solves then optimal binary values are fixed again. This procedure continues until upper bound is equal to lower bound. Final solution is a global solution for original problem. For very big dimensions, stopping criteria may be used to avoid extreme run time, such as maximum iteration and minimum thegap between lower and upper bounds. We present a pseudo-code for the proposed procedure as follows:

1. \( \text{iter} = 0 \) and \( p, f = \emptyset \), \( \text{LB} = -\infty \), \( \text{UB} = \infty \)
2. solve master problem
3. \( \text{iter} = \text{iter} + 1 \)
4. \( \text{fix} \_ W_{f(\text{iter})} = W_j \)
5. solve subproblem
6. if infeasible then \( f = f \cup \{ \text{iter} \} \)
7. else if \( p = p \cup \{ \text{iter} \} \)
8. \( \text{LB} = \max \left\{ \text{LB}, \text{obj}_{\text{sub}} - \sum Lo \text{ cost} \times \text{fix} \_ W_j \right\} \)
9. solve master problem
10. \( \text{UB} = \text{obj}_{\text{master}} \)
11. if \( \text{UB} = \text{LB} \) then \( \text{obj} = \text{obj}_{\text{master}} \)
12. else if go to step 3

4. Computational Results

4.1 Benders’ validation

We formulate the problem, and implement the proposed algorithm in GAMS 23.5 in relation with CPLEX® Solver version 12.2. In the first section, validity of GAMS codes are controlled via some test problems. All examples are generated randomly and they are solved by both regular GAMS and also Benders’ decomposition method. From Table 1, we can see that there is no difference between all the objective functions. Therefore, it is obvious that there is no gap when different problems are solved. This implies that the proposed Benders’ decomposition method as well as the regular GAMS solver is effective in solving the presented model.

Table 1

<table>
<thead>
<tr>
<th>M</th>
<th>N</th>
<th>L</th>
<th>No. of open DCs</th>
<th>No. of open plants</th>
<th>Total cost by Benders’</th>
<th>Total cost by GAMS</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>5</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>15221</td>
<td>15221</td>
</tr>
<tr>
<td>9</td>
<td>5</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>16044</td>
<td>16044</td>
</tr>
<tr>
<td>10</td>
<td>6</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>16215</td>
<td>16215</td>
</tr>
<tr>
<td>11</td>
<td>6</td>
<td>3</td>
<td>3</td>
<td>1</td>
<td>16899</td>
<td>16899</td>
</tr>
<tr>
<td>12</td>
<td>6</td>
<td>3</td>
<td>3</td>
<td>1</td>
<td>17451</td>
<td>17451</td>
</tr>
</tbody>
</table>

4.2 Sensitivity analysis

In this section, problems sets were generated randomly but systematically to capture a wide range of problems structures. The numbers of customer zones, potential DC and potential plants vary from 40 to 70, 6 to 12 and from
3 to 7. The demand mean requirements of customers were drawn from a uniform distribution between 10 and 100 which the demand has Poisson distribution function. If we let SUM represents total demand mean requirements, then the capacity of DCs were drawn from uniform distribution between 100 and SUM/2. And also the capacities of plants were drawn between SUM/2 and SUM+100.

The parameters $C_{ij}$, $C_{jk}$, $F_{ij}$, $G_{ij}$ and $h_{ij}$ follow uniform distributions which are in the below:

$C_{ij} \sim U(2, 10)$
$C_{jk} \sim U(2, 8)$
$F_{ij} \sim U(1500, 2500)$
$G_{ij} \sim U(5000, 7000)$
$h_{ij} \sim U(1, 8)$

Ordering cost was fixed to either 200 or 500. Any number in $Z_{ij}$ matrix can be 1 with probability 0.2 and can be 0 with probability 0.8.

The results per first set of experiments testing problems are reported in Table 1. The results are described by providing the number of user nodes $(|N|)$, the number of potential DC sites $(|M|)$, the number of potential plants $(|L|)$, the number of opened DCs and plants, total cost, and total loss of inventory cost. Finally, we reported the CPU times in seconds.

As we see in the Table 1 CPU times is sensitive to the number of customers and DCs. So, the more the number of customers and DCs, the more CPU time is needed to solve the problem.

Table 2: Experiments results for test problems

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th>No. of open DCs</th>
<th>No. of open plants</th>
<th>Total Cost</th>
<th>Inventory lost cost</th>
<th>$l_\alpha$</th>
<th>CPU time</th>
</tr>
</thead>
<tbody>
<tr>
<td>40</td>
<td>6</td>
<td>3</td>
<td>4</td>
<td>1</td>
<td>75000.22</td>
<td>792.96</td>
<td>0.84</td>
<td>207</td>
</tr>
<tr>
<td>40</td>
<td>6</td>
<td>3</td>
<td>4</td>
<td>1</td>
<td>75104.06</td>
<td>789.12</td>
<td>0.86</td>
<td>215</td>
</tr>
<tr>
<td>40</td>
<td>6</td>
<td>3</td>
<td>4</td>
<td>1</td>
<td>75109.33</td>
<td>785.28</td>
<td>0.88</td>
<td>202</td>
</tr>
<tr>
<td>40</td>
<td>6</td>
<td>3</td>
<td>3</td>
<td>1</td>
<td>73283.48</td>
<td>779.01</td>
<td>0.91</td>
<td>217</td>
</tr>
<tr>
<td>40</td>
<td>6</td>
<td>3</td>
<td>4</td>
<td>1</td>
<td>75128.54</td>
<td>775.67</td>
<td>0.93</td>
<td>225</td>
</tr>
<tr>
<td>40</td>
<td>6</td>
<td>3</td>
<td>4</td>
<td>1</td>
<td>75209.07</td>
<td>769.40</td>
<td>0.97</td>
<td>238</td>
</tr>
<tr>
<td>40</td>
<td>6</td>
<td>3</td>
<td>4</td>
<td>1</td>
<td>75698.11</td>
<td>755.27</td>
<td>0.99</td>
<td>235</td>
</tr>
<tr>
<td>50</td>
<td>8</td>
<td>3</td>
<td>4</td>
<td>1</td>
<td>110168.80</td>
<td>745.47</td>
<td>0.95</td>
<td>241</td>
</tr>
<tr>
<td>50</td>
<td>8</td>
<td>3</td>
<td>4</td>
<td>1</td>
<td>109999.60</td>
<td>1025.44</td>
<td>0.95</td>
<td>314</td>
</tr>
<tr>
<td>50</td>
<td>10</td>
<td>5</td>
<td>4</td>
<td>2</td>
<td>120958.60</td>
<td>984.29</td>
<td>0.90</td>
<td>465</td>
</tr>
<tr>
<td>50</td>
<td>10</td>
<td>5</td>
<td>4</td>
<td>1</td>
<td>119436.20</td>
<td>950.76</td>
<td>0.95</td>
<td>421</td>
</tr>
<tr>
<td>50</td>
<td>10</td>
<td>5</td>
<td>5</td>
<td>1</td>
<td>120335.54</td>
<td>343.80</td>
<td>0.97</td>
<td>463</td>
</tr>
<tr>
<td>50</td>
<td>10</td>
<td>5</td>
<td>4</td>
<td>1</td>
<td>119437.30</td>
<td>952.67</td>
<td>0.99</td>
<td>472</td>
</tr>
<tr>
<td>50</td>
<td>10</td>
<td>5</td>
<td>7</td>
<td>1</td>
<td>119739.50</td>
<td>343.80</td>
<td>0.99</td>
<td>512</td>
</tr>
<tr>
<td>50</td>
<td>10</td>
<td>5</td>
<td>6</td>
<td>1</td>
<td>133802.60</td>
<td>428.28</td>
<td>0.99</td>
<td>551</td>
</tr>
<tr>
<td>70</td>
<td>12</td>
<td>5</td>
<td>6</td>
<td>2</td>
<td>131962.70</td>
<td>2389.01</td>
<td>0.95</td>
<td>622</td>
</tr>
<tr>
<td>70</td>
<td>12</td>
<td>7</td>
<td>6</td>
<td>3</td>
<td>128432.10</td>
<td>789.96</td>
<td>0.99</td>
<td>642</td>
</tr>
</tbody>
</table>

Inventory theory confirms that once value of service level $(1-\alpha)$ increases, the costs related to inventory loss decreases, this way customers’ demand will be more satisfied and thus, the expected cost for inventory which is lost, decreases. As it can be found in Table 2, this concept will be verified clearly by local optimum solution. But the important notice is that by increasing $(1-\alpha)$, the transportation costs and holding costs will increase (because to have more service level, we must order more products and this leads to increased transportation and holding costs). So a trade of between them is necessary to find optimum value of $(1-\alpha)$. With comprehension above costs we can find best $(1-\alpha)$ with the minimum cost.

Figure 2 and Figure 3 illustrate the trend of total cost and loss of inventory cost for 2 problems with specific number of customers, DCs and plants regarding to the increasing $(1-\alpha)$. As it can be seen, loss of inventory cost decreases by increasing $(1-\alpha)$ but due to increasing transportation and holding cost with having a trade of between them, total cost may be increased or decreased.
5. Conclusion and Future Directions

In this paper, we introduced notation of distribution network in a supply chain system while we have service level constraint and customer’s demand is stochastic with Poisson distribution function. In the proposed model, we assumed that DCs have coverage radius restriction and also, in-depot transportation. We formulated the problem as a nonlinear integer program. The advantages of the proposed model are as follows: considering coverage radius which yields more flexibility for the model, considering service level constraint that by using it the required inventory can be computed automatically, it prevents inventory lost and finally all types of cost for inventory are consider in the model. Also, we presented a mixed integer programming model and proposed an exact solution procedure in regard to Benders’ decomposition method.

For future research we suggest three directions:
(a) We can study this model while \( Z_i \) is probability and this parameter does not have deterministic value. For example, it has a distribution probability function and based on it, \( Z_i \) gets value 0 or 1.
(b) Setting inventory capacity constraint is another development for this model.
(c) Applying the other ordering methods to calculate inventory costs in designing distribution networks.
(d) Aggregating this model with other assumption such as routing and considering time window can be an interesting development of the proposed model.

Acknowledgments

This research is supported by Grant of Islamic Azad University South Tehran branch by Faculty Research Support Fund from the Faculty of Engineering for project Development of hierarchical location model under stochastic situations considering risk optimization.

References
