Mixed-Integer Nonlinear Programming Models and Algorithms for Large-Scale Supply Chain Design with Stochastic Inventory Management

Fengqi You, Ignacio E. Grossmann*

Department of Chemical Engineering, Carnegie Mellon University
Pittsburgh, PA 15213, USA

Abstract

An important challenge for most chemical companies is to simultaneously consider inventory optimization and supply chain network design under demand uncertainty. This leads to a problem that requires integrating a stochastic inventory model with the supply chain network design model. This problem can be formulated as a large scale combinatorial optimization model that includes nonlinear terms. Since these models are very difficult to solve, they require exploiting their properties and developing special solution techniques to reduce the computational effort. In this work, we analyze the properties of the basic model and develop solution techniques for a joint supply chain network design and inventory management model for a given product. The model is formulated as a nonlinear integer programming problem. By reformulating it as a mixed-integer nonlinear programming (MINLP) problem and using an associated convex relaxation model for initialization, we first propose a heuristic method to quickly obtain good quality solutions. Further, a decomposition algorithm based on Lagrangean relaxation is developed for obtaining global or near-global optimal solutions. Extensive computational examples with up to 150 distribution centers and 150 retailers are presented to illustrate the performance of the algorithms and to compare them with the full-space solution.

---

* To whom all correspondence should be addressed. E-mail: grossmann@cmu.edu

1. Introduction

Due to increasing pressure for remaining competitive in the global market place, an emerging challenge for the process industries has become how to manage the inventories at the enterprise level so as to reduce costs and improve the customer service.[1, 2] A key challenge to achieve this goal is to integrate inventory management with supply chain network design decisions, so that decisions such as the number of inventory stocking locations and the associated amount of inventory can be determined simultaneously for lower costs and higher customer service level.

Although supply chain network design problems and the inventory management problems have been studied extensively in recent years,[3-8] most of the models consider inventory management and supply chain network design separately. On the other hand, there are related works on supply chain optimization that take into account the inventory costs, but consider inventory issues without detailed inventory management policies. In these models the safety stock level is given as a parameter, and usually treated as a lower bound of the total inventory level,[9-11] or considered as the inventory targets that would lead to some penalty costs if violated.[12-14] This approach cannot optimize the safety stock levels, especially when considering demand uncertainty.[15-17] Thus, it can only provide an approximation of the inventory cost, and therefore lead to suboptimal solutions. Jung et al.[18] introduce a simulation-optimization framework to estimate the optimal safety stock levels, but the supply chain design decisions are not jointly optimized.

Recently, Shen et al.[19] proposed a joint location-inventory model that integrates supply chain network design model with inventory management under demand uncertainty. In their work, the management of working inventory and safety stock are taken into account besides the distribution center location decisions. To solve the resulting nonlinear integer programming problem, the authors simplified the model by assuming that the uncertain demand in each retailer has the same variance-to-mean ratio. Based on this assumption, they reformulated the model as a set-covering problem and solved it with a branch-and-price algorithm. The proposed algorithm performs well for large scale problems. However, the assumption for identical variance-to-mean ratio might not provide a good approximation to real world problems because the demand uncertainties for each retailer may vary significantly. Thus, to allow the model to accommodate more general cases, an efficient algorithm is needed for the model without the simplifications.

Lagrangean relaxation and Lagrangean decomposition methods are recognized as efficient tools for solving large-scale optimization problems with "special" structures. The Lagrangean relaxation and subgradient optimization are discussed by Fisher.[20, 21] Later, Guignard and Kim[22] proposed the well-known Lagrangean decomposition method that yields stronger bounds than the Lagrangean relaxation algorithm. A large number of applications of Lagrangean-based algorithms for supply chain optimization and related problems have been reported in the past. Various Lagrangean-based heuristic algorithms for large scale facility location problems are discussed by Beasley.[23] Based on this work, Holmberg and Ling[24] proposed a novel Lagrangean heuristic method for location problems with staircase costs. Sridharan[25] implemented the Lagrangean relaxation method for the plant location problem with consideration of capacity issues. A Lagrangean relaxation and decomposition method for multiproduct tri-echelon supply chain design problem is proposed by Pirkul and Jayaraman.[26] Later, Klose[27] developed a relax-and-cut algorithm for the capacitated facility location problems. The proposed method yielded significant improvements in computational efficiency. van den Heever et al.[28] developed a Lagrangean heuristic method for the design and planning of offshore oil fields. A Lagrangean based temporal decomposition algorithm for supply chain planning was proposed by Jackson and Grossmann.[12] Recently, Neiro and Pinto[29] applied the Lagrangean based method to a petroleum supply chain planning model. The results showed that significant improvement in computational efficiency can be achieved by using Lagrangean decomposition.

The objective of this work is to develop effective algorithms for large-scale joint supply chain network design and inventory management problem for a given product. This work relies on the integer nonlinear programming model proposed by Shen et al.[19]. We first reformulate the model as a mixed-integer nonlinear programming (MINLP) model, and then solve it with different solution approaches, including a proposed heuristic method that relies on initialization from convex relaxations and a Lagrangean relaxation algorithm. The results from the full-scale solution and those from the various solution strategies are then compared and analyzed.

The rest of this paper is organized as follows. Some basic concepts of inventory management with risk pooling are discussed in Section 2. Section 3 presents the problem statement, while Section 4 provides a detailed description of the joint supply chain network design and inventory management model proposed by Shen et al.,[19] stating and defining the objective function and constraints of the model. The solution strategies including the MINLP reformulation and the Lagrangean relaxation algorithm are proposed in Section 5. Two small illustrative examples on a supply chain for liquid oxygen (LOX) are given in Section 6. Section 7 presents the comparison between different solution strategies and the full scale model, along with an analysis of the solution quality. Finally, Section 8 concludes on the performance of the proposed algorithm and the overall results.

2. Inventory Management Model with Risk Pooling

In this section, we briefly review some inventory management models that are related to the problem addressed in this work. Detailed discussion about inventory management models are given by Zipkin.[5]

Figure 1 shows the inventory profile in a distribution center (or any stocking facility) for a given product. As we can see, the inventory level decreases due to the customer demand, and increases when replenishments arrive. The reorder point is a specific inventory level. It means that each time when the inventory level goes down to the reorder point, a replenishment order will be placed. The time it requires from placing an order until the replenishment arrives at the distribution center is defined as the ordering lead time. Typically, the total inventory consists of two parts, working inventory and safety stock. The working inventory represents products that have been ordered from the supplier due to replenishment, but not yet shipped out of the distribution center to satisfy the demand. The safety stock is the inventory for buffering the system against stockouts due to the uncertain demands during the ordering lead time.

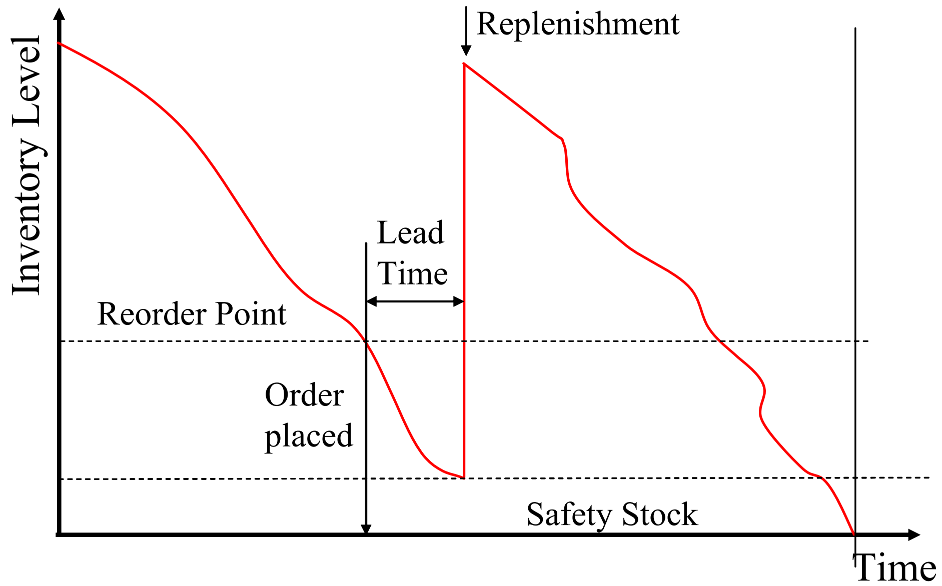


Figure 1 Inventory profile changing with time

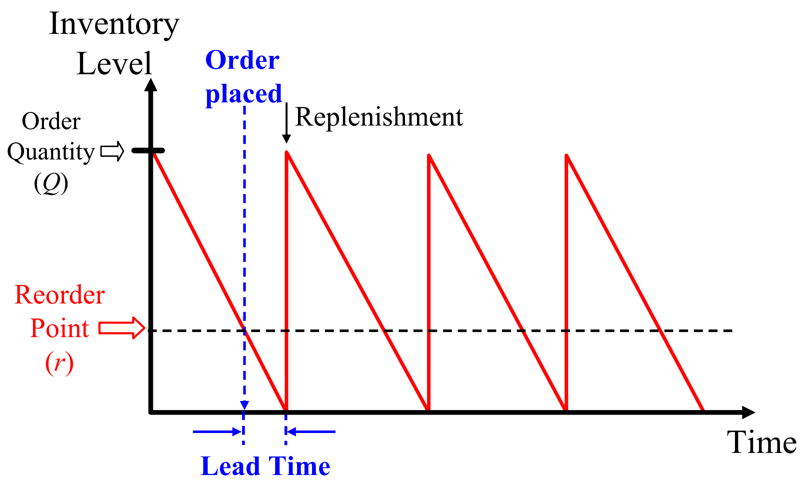


Figure 2 Inventory profile for deterministic demand with (Q, r) policy

A popular inventory control policy widely used in practice is the order quantity/reorder point (Q, r) inventory policy. When using this policy, each time when inventory level depletes to reorder point r, a fixed order quantity Q will be placed for replenishment. When the demand is deterministic with consistent demand rate, the inventory profile is a series of identical square triangle as given in Figure 2. Each of these square triangles has the same height (the order quantity Q), and the same width denoted as the replenishment interval. The optimal order quantity and replenishment interval for this deterministic demand case can be determined by using an economic order quantity (EOQ) model, which takes into account the trade-off between fixed ordering costs, transportation costs and working inventory holding costs (EOQ model formulation for our model is given in equation (3) in Section 3.2.). Although the EOQ model uses the deterministic demands, it has proved to provide very good approximations for working inventory costs of systems with (Q, r) policy under demand uncertainty.[30, 31] A common approach for the (Q, r) inventory model, as pointed out by Axsater,[30] is to first replace the stochastic demand with its mean value and then determine the optimal order quantity Q with the deterministic EOQ model, and finally find out the optimal reorder point under uncertain demand based on the order quantity.

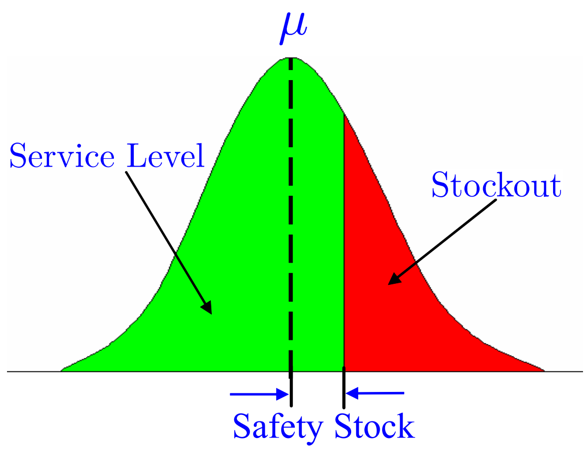


Figure 3 Safety stock and service level under normally distributed demand

A distribution center under demand uncertainty may not always have sufficient stock to handle the changing demand. If the reorder point (inventory level) is less than the demand during the order lead time, stockout may happen. *Type I service level* is defined as the probability that the total inventory on hand is more than the demand (as shown Figure 3). If the demand is normally distributed with mean μ and standard deviation σ and the ordering lead time is L, the optimal safety stock level to guarantee a service level α is $z_\alpha \sqrt{L}\sigma$, where z_α is a standard normal deviate such that $\Pr(z \le z_\alpha) = \alpha$.[5] We should note that the acceptable practice in this field is to assume a normal distribution of the demand, although of course other distribution functions can be specified.

To consider the total safety stock of an inventory system, Eppen[32] proposed the "risk pooling effect", which states that significant safety stock cost can be saved by grouping retailers. In particular, Eppen considered a single period problem with N retailers and one supplier. Each retailer i has normally distributed demand with mean μ_i and standard deviation σ_i, and the correlation coefficient of demand at retail i and j is ρ_{ij}. The order lead time from the supplier to all these retailers are the same and given as L. Eppen compared two operational modes of retailer supply chain: decentralized mode and centralized mode. In the decentralized mode, each retailer orders independently to minimize its own expected cost. Since in this mode the optimal safety stock in retailer i is $z_\alpha \sqrt{L}\sigma_i$, the total safety stock in the system is given by,

$$z_\alpha \sqrt{L} \sum_{i=1}^{N} \sigma_i \text{ .}$$

In the centralized mode, all the retailers are considered as a whole and a single quantity is ordered for replenishment, so as to minimize the total expected cost of the entire system. Since in the centralized mode all the retailers are grouped, and the demand at each retailer follows a normal distribution $N(\mu_i, \sigma_i^2)$, the total uncertain demand of the entire system during the order lead time will also follow a normal distribution with mean $L\sum_{i=1}^{N}\mu_i$ and standard deviation $\sqrt{L}\sqrt{\sum_{i=1}^{N}\sigma_i^2 + 2\sum_{i=1}^{N-1}\sum_{j=i+1}^{N}\sigma_i\sigma_j\rho_{ij}}$. Therefore, the total safety stock of the distribution centers in the centralized mode is,

$$z_\alpha \sqrt{L}\sqrt{\sum_{i=1}^{N}\sigma_i^2 + 2\sum_{i=1}^{N-1}\sum_{j=i+1}^{N}\sigma_i\sigma_j\rho_{ij}}$$

Thus, if the demands of all the N retailers are independent, the optimal safety stock can be expressed by $z_\alpha \sqrt{L}\sqrt{\sum_{i=1}^{N}\sigma_i^2}$, which is less than $z_\alpha \sqrt{L}\sum_{i=1}^{N}\sigma_i$. Eppen's simple model illustrates the potential saving in safety stock costs due to risk pooling.

In summary, for an inventory system including multiple distribution centers operating with (Q, r) policy and *Type I service level* under demand uncertainty, the total inventory cost consists of working inventory costs and safety stock costs. The optimal working inventory costs can be estimated with a deterministic EOQ model, and the safety stock costs can be reduced by risk pooling.

3. Problem Statement

We assume we are given a supply chain consisting of one or more suppliers and a set of retailers $i \in I$ (this can also be customers or markets, for convenience, we denote as "retailer" in the rest part of this paper unless specified), together with a number candidate sites for distribution centers $j \in J$. For an example of the network structure, see Figure 4). The locations of the supplier(s), potential distribution centers and the retailers are known and the distances between them are given. The replenishment lead time L of each distribution center is assumed to be the same for all the candidate distribution centers. This in turn means that the suppliers can be treated implicitly and lumped into one supplier. There is a fixed setup cost f_j when each distribution center is installed. Each retailer i has a normally distributed demand with mean μ_i and variance σ_i^2, which is independent of the other retailers' demands (The model can be easily extended to consider correlated demands in the retailers by modifying the safety stock terms as discussed at the end of Section 2). Each distribution center can serve more than one retailer, but each retailer should be only assigned to exactly one distribution center to satisfy the demand. Linear transportation costs are incurred for shipments from supplier to distribution center j with fixed cost g_j and unit cost a_j and from distribution center j to retailer i with unit cost d_{ij}. Most of the inventory in the network is held in the distribution centers where the inventory is managed with a (Q, r) policy with *type I service* .[5] Inventory costs are incurred at each distribution centers, and consist of both working inventory and safety stock. The retailers only maintain a very small amount of inventory whose costs are ignored.

The problem is to determine how many distribution centers (DCs) to install, where to locate them, which DCs to assign to each retailer, how often to reorder for replenishment at each DC, and what level of safety stock to maintain so as to minimize the total location, transportation, and inventory costs, while ensuring a specified level of service.

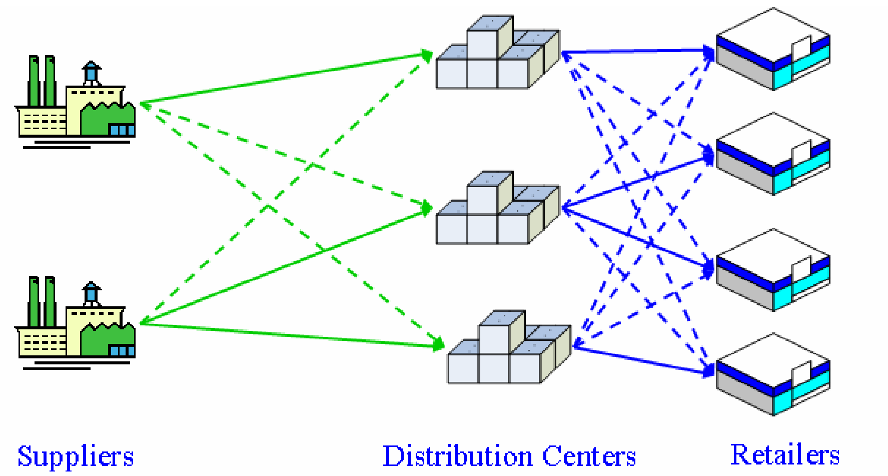


Figure 4 Supply chain network structure (three echelons)

4. Model Formulation

The joint supply chain network design and inventory management model of Shen et al. [19] is used as the basis for the present work, in which we will not rely on the assumption that each customer has identical variance-to-mean ratio. The joint location-inventory model is a nonlinear integer program that deals with the supply chain network design for a given product, and considers its detailed inventory management. The definition of sets, parameters, and variables of the model are as follows:

Sets/Indices

I Set of retailers indexed by i

J Set of candidate DC site indexed by j

Parameters

f_j Fixed cost (annual) of locating a DC at candidate site j

d_{ij} Unit transportation cost from DC j to retailer i

χ Days per year (to convert daily demand and variance values to annual costs)

μ_i Mean demand at retailer i (daily)

σ_i^2 Variance of demand at retailer i (daily)

F_j Fixed cost of placing an order from the supplier to the DC at candidate site j

g_j Fixed transportation cost from the supplier to the DC at candidate site j

a_j Unit transportation cost from the supplier to the DC at candidate site j

L Lead time from the supplier to the candidate DC sites (in days)

h Unit inventory holding cost

α Desired probability of retailer orders satisfied

β Weight factor assigned to transportation costs

θ Weight factor assigned to inventory costs

z_α Standard normal deviate such that $\Pr(z \le z_\alpha) = \alpha$

Decision Variables (0-1)

X_j 1 if we locate a DC in candidate site j, and 0 otherwise

Y_{ij} 1 if retailer i is served by the DC at candidate site j, and 0 otherwise

4.1. Objective Function

The objective of this model is to minimize the total weighted cost of the following items:

- fixed cost for locating facilities,
- transportation costs from DCs to retailers,
- fixed order placing costs, transportation costs from the supplier to DCs and the expected working inventory costs in the DCs,
- safety stock costs in DCs.

The facility location cost is given by,

$$\sum_{j \in J} f_j X_j \tag{1}$$

The product of yearly expected mean demand ($\chi\mu_i$) and the unit transportation cost (d_{ij}) leads to the annual DC to retailer transportation costs. If the retailer i is not served by the DC in candidate location j, the transportation cost is zero. Hence, the total expected transportation costs from DCs to retailers can be expressed as:

$$\sum_{j\in J}\sum_{i\in I}\chi d_{ij}\mu_i Y_{ij} \tag{2}$$

As all the retailers have stochastic demands and all the DCs manage the inventory using a (Q, r) policy with *Type I Service* constraint, the working inventory cost can be approximated with an economic order quantity model (EOQ) with very small error bound.[30, 31] Let n be the number of replenishments per year and D be the annual demand for the product. Thus, the annual costs of ordering, shipping and working inventory from the supplier to the DCs are approximated by:

$$Fn+\beta\left(g+a\frac{D}{n}\right)n+\theta\frac{hD}{2n} \tag{3}$$

The first term Fn is the total ordering cost per year. The second term is the annual transportation cost times the weighted factor (β), where (D/n) is the expected shipment size, and the shipping cost is given by a linear function $v(x)=g+ax$. The third term is the annual working inventory costs times the weighted factor (θ), where $D/(2n)$ is the average inventory level on hand. Considering (3) as a function of annual order number n, by setting the first order derivative to zero with respect to n, we can obtain the optimal order number $n=\sqrt{\theta hD/\left(2\left(F+\beta g\right)\right)}$. Therefore, by substituting into (3), the total optimal cost for replenishments, including ordering, transportation and working inventory holding cost is given by,

$$\beta aD+\sqrt{2\theta h\left(F+\beta g\right)D}\,. \tag{4}$$

Substituting the demand D with the annual expected demand of the product in each DC ($\sum_{i\in I}\chi\mu_i Y_{ij}$), the total replenishment costs for all the DCs can be expressed by,

$$\beta\sum_{j\in J}a_j\sum_{i\in I}\chi\mu_iY_{ij}+\sum_{j\in J}\sqrt{2\theta h\left(F_j+\beta g_j\right)\sum_{i\in I}\chi\mu_iY_{ij}}\ . \tag{5}$$

As the demand at each retailer follows a given normal distribution, let μ_i and σ_i^2 be the mean and variance of demand of the product at retailer i. Due to the risk-pooling effect,[32] the lead time demand at each DC is also normally distributed with a mean of $L\sum_{i\in S}\mu_i$ and a variance of $L\sum_{i\in S}\sigma_i^2$. Thus, the safety stock required in the DC at candidate location j to ensure that stockouts occur with a probability of α or less is,

$$z_\alpha\sqrt{L\sum_{i\in I}\sigma_i^2Y_{ij}}\ . \tag{6}$$

Therefore, the objective function of this model is given by

$$\begin{aligned}\text{Min: }&\sum_{j\in J}f_jX_j+\beta\sum_{j\in J}\sum_{i\in I}\chi d_{ij}\mu_iY_{ij}+\beta\sum_{j\in J}a_j\sum_{i\in I}\chi\mu_iY_{ij}\\&+\sum_{j\in J}\sqrt{2\theta h\left(F_j+\beta g_j\right)\sum_{i\in I}\chi\mu_iY_{ij}}+\theta hz_\alpha\sum_{j\in J}\sqrt{\sum_{i\in I}L\sigma_i^2Y_{ij}}\end{aligned} \tag{7}$$

where each term accounts for the fixed facility location cost, DC to retailer transportation costs, replenishment costs (including supplier to DC transportation costs, fixed ordering costs and working inventory costs) and safety stock costs.

4.2. Network Constraints

Two constraints are used to define the network structure. The first one is that each retailer i should be served by only one DC,

$$\sum_{j\in J}Y_{ij}=1, \qquad \forall i\in I\ . \tag{8}$$

The second constraint states that if a retailer i is served by the DC in candidate location j, the DC must exist,

$$Y_{ij}\le X_j, \qquad \forall i\in I,\ \forall j\in J\ . \tag{9}$$

Finally, all the decision variables are binary variables in this model:

$$X_j\in\{0,1\}, \qquad \forall j\in J \tag{10}$$

$$Y_{ij}\in\{0,1\}, \qquad \forall i\in I,\ \forall j\in J \tag{11}$$

4.3. INLP Model

Grouping the parameters, we can rearrange the objective function and formulate the problem to (**P0**) as the following integer nonlinear programming (INLP) problem:

$$(\mathbf{P0}) \quad \text{Min: } \sum_{j\in J}\left(f_j X_j + \sum_{i\in I}\hat{d}_{ij}Y_{ij} + K_j\sqrt{\sum_{i\in I}\mu_i Y_{ij}} + q\sqrt{\sum_{i\in I}\hat{\sigma}_i^2 Y_{ij}}\right) \tag{12}$$

$$\text{s.t.} \quad \sum_{j\in J} Y_{ij} = 1, \qquad \forall i \in I \tag{8}$$

$$Y_{ij} \le X_j, \qquad \forall i \in I,\ \forall j \in J \tag{9}$$

$$X_j \in \{0,1\}, \qquad \forall j \in J \tag{10}$$

$$Y_{ij} \in \{0,1\}, \qquad \forall i \in I,\ \forall j \in J \tag{11}$$

where

$$\hat{d}_{ij} = \beta\chi\mu_i\left(d_{ij} + a_j\right)$$

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

$$q = \theta h z_\alpha$$

$$\hat{\sigma}_i^2 = L\sigma_i^2$$

5. Solution Approach

The joint supply chain network design and inventory management model ((8)-(12)) is a nonlinear integer program where all the decision variables are binary variables. Besides its combinatorial nature, the nonlinear terms are nonconvex which make the optimization model very difficult to solve. In order to address this problem, previous researchers[19, 33] have simplified the model by assuming that the variance-to-mean ratio at all the retailers are identical, but in the real world this ratio for each retailer may vary from others, and thus an efficient algorithm is required to solve the model (**P0**) without the aforementioned assumption so as to provide a good approximation for real cases. In the next section, we reformulate the INLP model (**P0**) as a mixed-integer nonlinear programming (MINLP) problem with fewer 0-1 variables and solve it with different solution approaches, including a heuristic method to obtain "good quality" solutions very quickly, and a Lagrangean relaxation algorithm for obtaining global or near-global optimal solutions.

5.1. MINLP Formulation

The original INLP model (**P0**) is very difficult to solve for large instances due to the potentially large number of binary variables (see Table 2 in Section 7 for examples). As shown in the proposition below, the assignment variables (Y_{ij}) in the model can be relaxed as continuous variables without changing the optimal integer solution. This allows us to reformulate (**P0**) as a MINLP problem with fewer 0-1 variables and most of them appearing in linear form.

Proposition 1. *The continuous variables Y_{ij} yield 0-1 integer values when (**P0**) is globally optimized or locally optimized for fixed 0-1 value for X_j.*

Proposition 1 means that the following problem (**P1**) yields integer values on the assignment variables Y_{ij} when it is globally optimized or locally optimized for a fixed 0-1 integer value for X_j.

(**P1**) Min: $$\sum_{j \in J} \left(f_j X_j + \sum_{i \in I} \hat{d}_{ij} Y_{ij} + K_j \sqrt{\sum_{i \in I} \mu_i Y_{ij}} + q \sqrt{\sum_{i \in I} \hat{\sigma}_i^2 Y_{ij}} \right) \tag{13}$$

s.t. $$\sum_{j \in J} Y_{ij} = 1, \qquad \forall i \in I. \tag{8}$$

$$Y_{ij} \le X_j, \qquad \forall i \in I, \ \forall j \in J. \tag{9}$$

$$X_j \in \{0,1\}, \qquad \forall j \in J \tag{10}$$

$$Y_{ij} \ge 0, \qquad \forall i \in I, \ \forall j \in J. \tag{14}$$

The proof, which is given in Appendix A, is based on the fact that for fixed X_j problem (**P1**) is a concave minimization problem defined over a polyhedron, and for which local and global solution for fixed integer values of X_j, yield integer values for the continuous variables Y_{ij}.

Proposition 1 allows us to solve the MINLP model (**P1**) instead of the INLP model (**P0**) significantly reducing the computational effort. It is interesting to note that if we set the unit inventory holding cost $h = 0$, the square root terms in the objective function (13) can be removed and problem (**P1**) reduces to the widely studied "Uncapacitated Facility Location" (UFL) problem,[3, 4, 34, 35] which is known to exhibit integer solutions for relaxed variables Y_{ij}. Furthermore, this problem is also known to be solvable through its LP relaxation for most instances.

(**P1**) is an MINLP problem with linear constraints and a nonlinear objective function including nonconvexities in the continuous variables. Optimization methods that can be used for obtaining the global optimal solution of problem (**P1**) include the branch and reduce method,[36, 37] the α-BB method,[38] the spatial branch and bound search method for bilinear and linear fractional terms[39, 40] and the outer-approximation method by Kesavan et al.[41] All these methods rely on a branch and bound solution procedure. The difference among these methods lies on the definition of the convex envelopes for computing the lower bound, and on how to perform the branching on the discrete and continuous variables. The global optimization solver that is commercially available is BARON,[42] which implements a branch-and-reduce solution method.

Since a global optimization algorithm can be expensive, another alternative is to use an MINLP method that relies on the assumption that the functions are convex. Although in this case global optimality cannot be guaranteed, the solutions can be obtained much faster, because a local optimal solution can be efficiently be found for a fixed value of the integer variables (optimal or near optimal). A general review of these MINLP methods is given in Grossmann.[43] Methods include the branch and bound method,[44] Generalized Benders Decomposition,[45] Outer-Approximation,[46, 47] LP/NLP based branch and bound,[48] and Extended Cutting Plane Method.[49] A number of computer codes are available that implement these methods. The program DICOPT[47] is an MINLP solver that is based on the Outer Approximation algorithm,[46] and is available in the modeling system GAMS.[50] It should be noted that this code has a heuristic termination criterion for nonconvex problems. The code α-ECP implements the extended cutting plane method by Westerlund and Pettersson.[49] Codes that implement the branch and bound method include the code MINLP_BB[44] available in AMPL, and the program SBB which is also available in GAMS. Recently, the open source MINLP solver Bonmin,[51] which is part of the COIN-OR project,[52] implements an extension of the branch-and-cut outer-approximation algorithm that was proposed by Quesada and Grossmann,[48] as well as the branch and bound and outer-approximation method.

5.2. MINLP Reformulation

In order to improve the computational efficiency of solving the MINLP model (**P1**) with the above cited solvers, we present in this section a reformulation of (**P1**).

The square root term in the objective function of (**P1**) can give rise to difficulties in the optimization procedure. When the DC in location j is not selected, both square root terms would take a value of zero, which leads to unbounded gradients in the NLP optimization and hence numerical difficulties. Thus, we reformulate the model in order to eliminate the square root terms. We first introduce two sets of non-negative continuous variables, $Z1_j$ and $Z2_j$, to represent the square-root terms in the objective function:

$$Z1_j^{\ 2} = \sum_{i \in I} \mu_i Y_{ij} , \qquad \forall j \in J \tag{15}$$

$$Z2_j^{\ 2} = \sum_{i \in I} \hat{\sigma}_i^2 Y_{ij} , \qquad \forall j \in J \tag{16}$$

$$Z1_j \geq 0 , \;\; Z2_j \geq 0 , \quad \forall j \in J \tag{17}$$

Because the non-negative variables $Z1_j$ and $Z2_j$ are introduced in the objective function with positive coefficients, and this problem is a minimization problem, (15) and (16) can be further relaxed as the following inequalities,

$$-Z1_j^{\ 2} + \sum_{i \in I} \mu_i Y_{ij} \leq 0 , \qquad \forall j \in J \tag{18}$$

$$-Z2_j^{\,2} + \sum_{i \in I} \hat{\sigma}_i^2 Y_{ij} \le 0, \qquad \forall j \in J \tag{19}$$

Thus, the reformulated model can then be expressed as the following MINLP problem denoted as (**P2**),

$$(\mathbf{P2}) \qquad \text{Min: } \sum_{j \in J} \left(f_j X_j + \sum_{i \in I} \hat{d}_{ij} Y_{ij} + K_j Z1_j + qZ2_j \right) \tag{20}$$

$$\text{s.t. } \sum_{j \in J} Y_{ij} = 1, \qquad \forall i \in I. \tag{8}$$

$$Y_{ij} \le X_j, \qquad \forall i \in I, \ \forall j \in J. \tag{9}$$

$$-Z1_j^{\,2} + \sum_{i \in I} \mu_i Y_{ij} \le 0, \qquad \forall j \in J \tag{18}$$

$$-Z2_j^{\,2} + \sum_{i \in I} \hat{\sigma}_i^2 Y_{ij} \le 0, \qquad \forall j \in J \tag{19}$$

$$X_j \in \{0,1\}, \qquad \forall j \in J \tag{10}$$

$$Y_{ij} \ge 0, \qquad \forall i \in I, \ \forall j \in J. \tag{14}$$

$$Z1_j \ge 0, \ Z2_j \ge 0, \quad \forall j \in J \tag{17}$$

(**P2**) can be trivially shown to be equivalent to (**P1**) but with linear objective function and quadratic terms in the constraints (18) and (19). As shown in Appendix A, the following property can be established for problem (**P2**).

Proposition 2. *The global optimal solution of problem (**P2**), or a local optimal solution with fixed 0-1 value for* X_j*, has all the continuous variables* Y_{ij} *take on integer value (0 or 1).*

5.3. Heuristic Algorithm

Since problem (**P2**) is a nonconvex problem, the solution is highly dependent of the starting point when using an MINLP solver that relies on convexity assumption. To obtain a “good” feasible starting point, we first relax the nonconvex nonlinear constraints (18) and (19) in (**P2**) by replacing the concave terms with their corresponding secants, which represent the convex envelopes[53] of these functions.

From (15) and (16), it is easy to see that the lower bounds $Z1_j$ and $Z2_j$ are both 0, and their upper bounds are $\sqrt{\sum_{i\in I}\mu_i}$ and $\sqrt{\sum_{i\in I}\hat{\sigma}_i^2}$ repectively.

Therefore, the secant of (18) is given by,

$$-\sqrt{\sum_{i\in I}\mu_i}\cdot Z1_j+\sum_{i\in I}\mu_i Y_{ij}\le 0,\quad \forall j\in J \tag{21}$$

Similarly, the secant of (19) is given by,

$$-\sqrt{\sum_{i\in I}\hat{\sigma}_i^2}\cdot Z2_j+\sum_{i\in I}\hat{\sigma}_i^2 Y_{ij}\le 0,\ \forall j\in J \tag{22}$$

In this way the convex relaxation of model (**P2**) can be formulated as Problem (**P3**):

(**P3**) Min: $$\sum_{j\in J}\left(f_j X_j+\sum_{i\in I}\hat{d}_{ij}Y_{ij}+K_j Z1_j+qZ2_j\right) \tag{20}$$

s.t. $$\sum_{j\in J}Y_{ij}=1,\qquad \forall i\in I. \tag{8}$$

$$Y_{ij}\le X_j,\qquad \forall i\in I,\ \forall j\in J. \tag{9}$$

$$-\sqrt{\sum_{i\in I}\mu_i}\cdot Z1_j+\sum_{i\in I}\mu_i Y_{ij}\le 0,\quad \forall j\in J \tag{21}$$

$$-\sqrt{\sum_{i\in I}\hat{\sigma}_i^2}\cdot Z2_j+\sum_{i\in I}\hat{\sigma}_i^2 Y_{ij}\le 0,\ \forall j\in J \tag{22}$$

$$X_j\in\{0,1\},\qquad \forall j\in J \tag{10}$$

$$Y_{ij}\ge 0,\qquad \forall i\in I,\ \forall j\in J. \tag{14}$$

$$Z1_j\ge 0,\ Z2_j\ge 0,\quad \forall j\in J \tag{17}$$

(**P3**) is a mixed-integer linear programming (MILP) problem which is the convex relaxation of problem (**P2**). The optimal solution of variables X_j and Y_{ij} of problem (**P3**) is a feasible solution of problem (**P2**) due to the linear constraints (8) and (9), and it can provide an initial point before solving (**P2**) with an MINLP solver. In this way, we can greatly speed up the computation and enhance the likelihood of obtaining a near-optimal solution of model (**P2**). In summary, the heuristic algorithm for obtaining a good quality solution with reasonable computational effort by using MINLP solvers that rely on convexity assumptions is as follows:

Algorithm 1: (Heuristic Algorithm)

Step 1: Solve the MILP model (**P3**).

Step 2: Use the optimal values of variables X_j and Y_{ij} obtained from Step 1 as the starting point, and solve problem (**P2**) with an MINLP solver that relies on convexity assumptions (such as DICOPT, SBB, α-ECP, MINLP_BB, Bonmin, etc.) for obtaining a near-optimal solution.

Note that if we solve problem (**P2**) with Algorithm 1 by using an MINLP solver that relies on convexity assumptions, the optimal solution may not be globally optimal. However, the optimal solution still has all the Y_{ij} variables at integer values based on Proposition 1 (see Appendix A for details). Furthermore, the solution obtained by using heuristic Algorithm 1 for problem (**P2**) is also a feasible solution of problem (**P1**).

5.4. A Lagrangean Relaxation Algorithm

In order to obtain potentially better solutions, we propose a Lagrangean relaxation algorithm for obtaining global optimal or near global optimal solutions of model (**P2**).

5.4.1. The Decomposition Procedure

In the Lagrangean relaxation algorithm, we use a "spatial" decomposition scheme by dualizing the assignment constraints (8) in (**P2**) using the Lagrangean multipliers λ_i, which is similar to the works by Beasley[23] and Daskin et al.[33] As a result, we obtain the following relaxed problem (denoted by $\mathbf{P}(\boldsymbol{\lambda})$),

$$(\mathbf{P}(\boldsymbol{\lambda}))\quad V = \text{Min: } \sum_{j \in J}\left(f_j X_j + \sum_{i \in I}\left(\hat{d}_{ij} - \lambda_i\right) Y_{ij} + K_j Z1_j + qZ2_j \right) + \sum_{i \in I} \lambda_i \tag{23}$$

$$\text{s.t.}\quad Y_{ij} \le X_j, \qquad \forall i \in I,\ \forall j \in J \tag{9}$$

$$-Z1_j^{\,2} + \sum_{i \in I} \mu_i Y_{ij} \le 0, \qquad \forall j \in J \tag{18}$$

$$-Z2_j^{\ 2} + \sum_{i \in I} \hat{\sigma}_i^2 Y_{ij} \le 0, \qquad \forall j \in J \tag{19}$$

$$X_j \in \{0,1\}, \qquad \forall j \in J \tag{10}$$

$$Y_{ij} \ge 0, \qquad \forall i \in I,\ \forall j \in J. \tag{14}$$

$$Z1_j \ge 0,\ Z2_j \ge 0, \quad \forall j \in J \tag{17}$$

where V is the objective function value. Next, we observe that ($\mathbf{P(\lambda)}$) can be decomposed into $|J|$ subproblems, one for each candidate DC site $j \in J$, where each one is denoted by ($\mathbf{P}_j\mathbf{(\lambda)}$) and is shown for a specific subproblem for candidate DC site j^* as follows:

($\mathbf{P}_{j^*}\mathbf{(\lambda)}$) $$V_{j^*} = \text{Min: } f_{j^*}X_{j^*} + \sum_{i \in I}\left(\hat{d}_{ij^*} - \lambda_i\right)Y_{ij^*} + K_{j^*}Z1_{j^*} + qZ2_{j^*} \tag{24}$$

s.t. $$Y_{ij^*} \le X_{j^*}, \qquad \forall i \in I$$

$$-Z1_{j^*}^{\ 2} + \sum_{i \in I} \mu_i Y_{ij^*} \le 0$$

$$-Z2_{j^*}^{\ 2} + \sum_{i \in I} \hat{\sigma}_i^2 Y_{ij^*} \le 0$$

$$Y_{ij^*} \ge 0, \qquad \forall i \in I$$

$$X_{j^*} \in \{0,1\}$$

$$Z1_{j^*} \ge 0,\ Z2_{j^*} \ge 0$$

Subproblem ($\mathbf{P}_j\mathbf{(\lambda)}$) has one binary variable (X_{j^*}), $|I|+2$ continuous variables ($Z1_{j^*}$, $Z2_{j^*}$, Y_{ij^*}) and $2|I|+2$ constraints. Because we have $|J|$ subproblems ($\mathbf{P}_j\mathbf{(\lambda)}$), and one for each candidate DC site $j \in J$, we call it a "spatial" decomposition scheme, i.e. decomposition by the spatial structure of the supply chain network.[12, 26] Let V_j denote the globally optimal objective function value of problem ($\mathbf{P}_j\mathbf{(\lambda)}$). As a result of the decomposition procedure, the globally optimal objective function value of ($\mathbf{P(\lambda)}$), which corresponds to a lower bound of problem (**P2**), can be calculated by:

$$V = \sum_{j \in J} V_j + \sum_{i \in I} \lambda_i \quad . \tag{25}$$

For each fixed value of the Lagrangean multipliers λ_i, we solve problem ($\mathbf{P}_j(\boldsymbol{\lambda})$) by globally minimizing (24) for each candidate DC location j (e.g. using BARON). Then, based on (25), the optimal objective function value of problem ($\mathbf{P}(\boldsymbol{\lambda})$) can be calculated for each fixed value of λ_i. Using a standard subgradient method[20, 21] to update the Lagrangean multiplier λ_i, the algorithm iterates until a preset optimality tolerance is reached.

5.4.2. Lagrangean Relaxation Subproblems

In each iteration with fixed values of the Lagrange multipliers λ_i, the design variables (X_j) are optimized separately in each subproblem ($\mathbf{P}_j(\boldsymbol{\lambda})$) in the aforementioned decomposition procedure. For each subproblem ($\mathbf{P}_j(\boldsymbol{\lambda})$), we can observe that the objective function value of ($\mathbf{P}_j(\boldsymbol{\lambda})$) is 0 if and only if $X_j = 0$ (i.e. we do not select DC j). In other words, there is a feasible solution that leads to the objective function value of subproblem ($\mathbf{P}_j(\boldsymbol{\lambda})$) equal to 0. Therefore, the globally minimum objective function value of subproblem ($\mathbf{P}_j(\boldsymbol{\lambda})$) should be less than or equal to zero. Given this observation, it is *possible* that under some value of λ_i (such as $\lambda_i = 0$, $i \in I$) the optimal objective function values for all the subproblem ($\mathbf{P}_j(\boldsymbol{\lambda})$) are 0 (i.e., $X_j = 0$, $j \in J$, we do not select any DC). However, the original assignment constraint (8) implies a redundant constraint that at least one DC should be selected to meet the demands, i.e.

$$\sum_{j \in J} X_j \geq 1 \, . \tag{26}$$

Once constraint (8) is relaxed, constraint (26) becomes "not redundant" and should be taken into account in the algorithm.[23, 26] To satisfy the constraint (26) in the Lagrangean relaxation procedure, we make the following modifications to the aforementioned step of solving problem $\mathbf{P}_j(\boldsymbol{\lambda})$ for each candidate DC location j.

First, consider the problem ($\mathbf{PR}_j(\boldsymbol{\lambda})$), which is actually a special case of ($\mathbf{P}_j(\boldsymbol{\lambda})$) when $X_j = 1$. The formulation for a specific j^* is given as:

$$(\mathbf{PR}_{j^*}(\boldsymbol{\lambda})) \quad \hat{V}_{j^*} = \text{Min: } f_{j^*} + \sum_{i \in I} \left(\hat{d}_{ij^*} - \lambda_i \right) Y_{ij^*} + K_{j^*} Z1_{j^*} + qZ2_{j^*} \tag{27}$$

$$\begin{aligned} \text{s.t. } & Y_{ij^*} \le 1, \qquad \forall i \in I \\ & -Z1_{j^*}^{\;2} + \sum_{i \in I} \mu_i Y_{ij^*} \le 0 \\ & -Z2_{j^*}^{\;2} + \sum_{i \in I} \hat{\sigma}_i^2 Y_{ij^*} \le 0 \\ & Y_{ij^*} \ge 0, \qquad \forall i \in I \\ & Z1_{j^*} \ge 0, \;\; Z2_{j^*} \ge 0 \end{aligned}$$

where $\hat{V}_j$ is denoted as the globally optimal objective function value of the problem ($\mathbf{PR}_j(\boldsymbol{\lambda})$).

Note that the X_j variable does not appear in subproblem ($\mathbf{PR}_j(\boldsymbol{\lambda})$). Therefore, the minimum objective function value of subproblem ($\mathbf{PR}_j(\boldsymbol{\lambda})$) is equal to the minimum objective function value of problem ($\mathbf{P}_j(\boldsymbol{\lambda})$) when $X_j = 1$. However, it is not always the same as the globally minimum objective function value of problem ($\mathbf{P}_j(\boldsymbol{\lambda})$), because ($\mathbf{P}_j(\boldsymbol{\lambda})$) could be globally optimal when $X_j = 0$,

For each fixed value of the Lagrange multiplier λ_i, if the globally minimum objective function value of the Lagrange subproblem ($\mathbf{PR}_j(\boldsymbol{\lambda})$) is negative, it means that when $X_j = 1$ the minimum objective function value of problem ($\mathbf{P}_j(\boldsymbol{\lambda})$) is negative. Because we know when $X_j = 0$ the objective function value of problem ($\mathbf{P}_j(\boldsymbol{\lambda})$) is 0, it follows that under this value of the Lagrange multiplier, the globally minimum objective function value of problem ($\mathbf{P}_j(\boldsymbol{\lambda})$) is the same as the minimum objective function value of problem ($\mathbf{PR}_j(\boldsymbol{\lambda})$), which is a negative value. Therefore, it is optimal to have $X_j = 1$ under this value of the Lagrange multiplier.

On the other hand, if the minimum objective function value of problem ($\mathbf{PR}_j(\boldsymbol{\lambda})$) is positive, it means that when $X_j = 1$, the optimal objective function value of problem ($\mathbf{P}_j(\boldsymbol{\lambda})$) could not be negative. Thus the optimal objective function value of problem ($\mathbf{P}_j(\boldsymbol{\lambda})$) would be 0 when $X_j = 0$ (because if $X_j = 1$ the minimum objective function value would be positive, as given in the objective function value of problem ($\mathbf{PR}_j(\boldsymbol{\lambda})$).

A possible extreme case is that the minimum objective function value of all the Lagrangean subproblems ($\mathbf{PR}_j(\boldsymbol{\lambda})$) are positive (for example, $\lambda_i = 0$, $i \in I$). In this case, it means that the globally minimum objective function value of problem ($\mathbf{P}_j(\boldsymbol{\lambda})$) are all 0, i.e. we do not select any DC. To satisfy the implied constraint (26) that we need to select at least one DC, we just install the DC j with smallest objective function value, though this value is positive. By using the relationship between problem ($\mathbf{PR}_j(\boldsymbol{\lambda})$) and ($\mathbf{P}_j(\boldsymbol{\lambda})$), we can solve ($\mathbf{PR}_j(\boldsymbol{\lambda})$) instead of ($\mathbf{P}_j(\boldsymbol{\lambda})$) for equivalent optimality.

Therefore, the algorithm for solving the Lagrangean relaxation subproblems is as follows. For each fixed value of λ_i, we solve $\mathbf{PR}_j(\boldsymbol{\lambda})$ for every candidate DC location j. Then select the DCs in candidate location j (i.e. let $X_j = 1$), for which $\hat{V}_j \le 0$. For all the remaining DCs for which $\hat{V}_j > 0$, we do not select them and set $X_j = 0$. On the other hand, if all the $\hat{V}_j > 0$, $\forall j \in J$, we select only one DC with the minimum $\hat{V}_j$, i.e. $X_{j*} = 1$ for the $j*$ such that $\hat{V}_{j*} = \min_{j \in J}\{\hat{V}_j\}$.

By doing this at each iteration of the Lagrangean relaxation (for each value of the multiplier λ_i), we ensure that the optimal solution always satisfies $\sum_{j \in J} X_j \ge 1$. Thus the globally optimal objective function of ($\mathbf{P}(\boldsymbol{\lambda})$) can be recalculated as:

$$V = \sum_{j \in J, X_j = 1} \hat{V}_j + \sum_{i \in I} \lambda_i \quad . \tag{28}$$

5.4.3. Obtaining Feasible Solutions

As the original model (**P2**) has very few constraints, there are several methods to obtain a feasible solution for the problem.

The initial feasible solution can be obtained with the following two methods:

The first method is to select a DC, and then assign all the retailers to this DC, i.e. pick up a $j^* \in J$, and let $X_{j^*} = 1$, $Y_{ij^*} = 1, \forall i \in I$. This method provides a simple way for obtaining a feasible solution and the resulting objective function value provides a valid upper bound of the global optimal objective function value.

The second method is to solve the problem (**P2**) with an MINLP solver, possibly using Algorithm 1 to obtain a near optimal solution, which is also a feasible solution of the original problem. This usually provides a "tighter" upper bound than the one obtained with the first method.

To obtain a feasible solution during the iterations, we first fix the values of the design variables (X_j) at the optimal values of the Lagrangean relaxation subproblems, and then solve the original model (**P2**) with a nonlinear programming (NLP) solver (not necessarily a global solver). The optimal values of the assignment variables (Y_{ij}) in the Lagrangean relaxation subproblems are used as the initial values in the nonlinear optimization procedure. Nonlinear solvers such as MINOS, CONOPT, SNOPT, KNITRO, IPOPT. can be used in this step.

Note that solving (**P2**) with fixed values of X_j variables using a local or global NLP solver guarantees that the feasible solutions generated in this step have all the Y_{ij} variables at integer values based on Proposition 2 (see Appendix A for details).

5.4.4. The Solution Algorithm

To summarize, the solution algorithm is as follows:

Algorithm 2: (Lagrangean Relaxation Algorithm)

Step 1: (Initialization) for the initial value of the multiplier λ^1 of constraint (8) use an arbitrary guess, or the multiplier values corresponding to a local optimum of the NLP relaxation of model (**P2**). Let the incumbent upper bound be $UB = +\infty$, lower bound be $LB = -\infty$ and iteration number be $t = 1$. Set the step length parameter $\theta = 2$.

Step 2: Solve the modified Lagrangean relaxation program ($\mathbf{PR}_j(\boldsymbol{\lambda}^t)$) with fixed Lagrangean multiplier vector λ^t for all the j using a global optimization solver (e.g. BARON). Denote the optimal objective function value as $\hat{V}_j\ (\lambda^t)$ and the optimal solutions as $\hat{Y}_{ij}(\lambda^t)$.

If $\hat{V}_j(\lambda^t) > 0$, $\forall j \in J$, let $X_{j*}(\lambda^t) = 1$ for the $j*$ such that $\hat{V}_{j*}(\lambda^t) = \min_{j \in J}\{\hat{V}_j(\lambda^t)\}$.

Else, let $X_j(\lambda^t) = 1$ for all j with $\hat{V}_j(\lambda^t) \le 0$, and $X_j(\lambda^t) = 0$ for all j such that $\hat{V}_j(\lambda^t) > 0$. Calculate $V(\lambda^t) = \sum_{j \in J, X_j(\lambda^t)=1} \hat{V}_j + \sum_{i \in I} \lambda_i^t$.

If $V(\lambda^t) > LB$, update the lower bound by setting $LB = V(\lambda^t)$,.

If more than 2 iterations of the subgradient procedure[20] are performed without an increment of LB, then halve the step length parameter by setting $\theta = \frac{\theta}{2}$.

Step 3: Fixing the design variable values as $X_j = X_j(\lambda^t)$ and using $\hat{Y}_{ij}(\lambda^t)$ as the initial values of the assignment variables Y_{ij}, solve problem ($\mathbf{P}(\boldsymbol{\lambda}^t)$) in the reduced space with fixed λ^t and $X_j(\lambda^t)$ using an NLP solver (local or global). Denote the optimal solution as $Y_{ij}(\lambda^t)$ and the optimal objective function value as $\overline{V}(\lambda^t)$.

If $\overline{V}(\lambda^t) < UB$, update the upper bound by setting $UB = \overline{V}(\lambda^t)$.

Step 4: Calculate the subgradient (G_i) using

$$G_i^t = 1 - \sum_{j \in J} \hat{Y}_{ij}(\lambda^t),\ i \in I \tag{29}$$

Compute the step size T, [20, 21]

$$T^t = \frac{\theta \cdot (UB - LB)}{\sum_{i \in I} (G_i^t)^2} \tag{30}$$

Update the multipliers:

$$\lambda^{t+1} = \max\{0, \lambda^t + T^t \cdot G^t\} \tag{31}$$

<u>Step 5:</u> If $gap = \frac{UB - LB}{UB} < tol$ (e.g. 10^{-5}), or $\left\| \lambda^{t+1} - \lambda^t \right\|^2 < tol$ (e.g. 10^{-3}) or the maximum number of iterations has been reached, set UB as the optimal objective function value, and set $X_j(\lambda^t)$ and $Y_{ij}(\lambda^t)$ as the optimal solution.

Else, increment t as $t+1$, go to Step 2.

We should note that the above algorithm is guaranteed to provide rigorous lower bounds in Step 2 since the subproblems are globally optimized. Also, the feasible solution generated in Step 3 has all the Y_{ij} variables at integer values as we mentioned in Section 5.3.3. Thus, the solution obtained by using this algorithm for problem (**P2**) is also a feasible solution of problem (**P1**). Due to the duality gap, the above algorithm must be stopped after a finite number of iterations. As will be shown in the computational results, the duality gaps are quite small.

6. Illustrative Example

To illustrate the application of this model, we consider a small illustrative example for the supply chain of liquid oxygen (LOX) consisting of one plant, three potential DCs and six customers as given in Figure 5. The tri-echelon "plant-DC-customer" supply chain is similar to the "supplier-DC-retailer" network that we discussed before, and the joint supply chain design and inventory management model can be also used to minimize total network design, transportation and inventory costs.

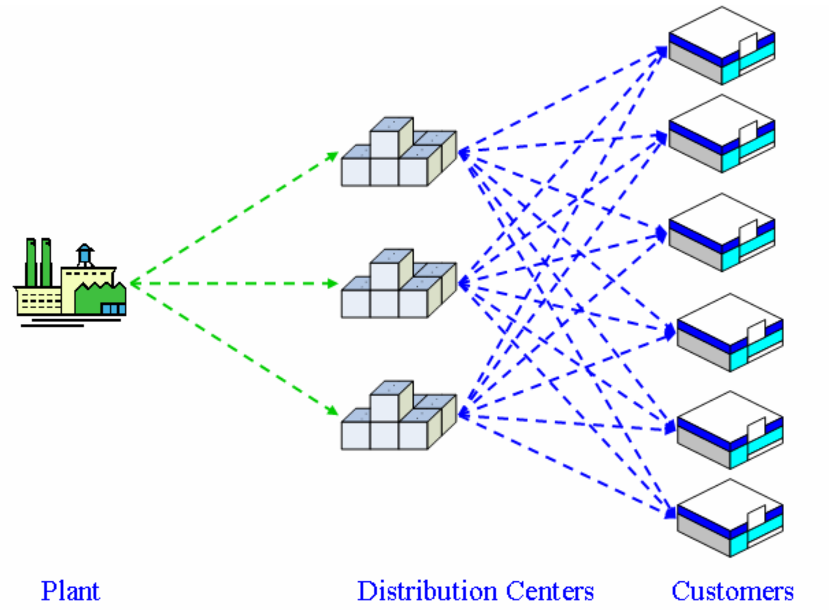


Figure 5 LOX supply chain network superstructure for the illustrative example

6.1. Illustrative Example for Industrial Application

We first consider an instance to illustrate the application of the joint supply chain design and inventory management model. In this instance, the fixed cost to install a DC (f_j) is \$10,000/year, and the fixed cost for ordering from supplier (F_j) is \$100/ replenishment. The order lead time for all the DCs are 7 days, and we consider 97.5% service level, thus the associated service level parameter z_α is 1.96. We consider 365 days in a year, and the annual inventory holding for LOX is \$3.65/Liter, i.e. daily inventory holding cost is \$0.01/Liter. Both weight parameters β and θ are set to 1. The remaining data for demand uncertainty and transportation costs are given in Table B1, B2 and B3 in Appendix B.

We solve model (**P2**) directly to obtain the global optimum by using the BARON solver with GAMS,[50] because the problem only includes 3 binary variables, 24 continuous variables and 30 constraints. The resulting optimal supply chain is given in Figure 6. We can see that only two DCs are installed, and they both serve the nearest three customers. The optimal replenishment number for DC1 is around 44 times, and for DC3 is around 57 times. This means that 108,770 liters of LOX are shipped from the plant to DC1 in 44 shipments, i.e. roughly one shipment every eight days, and 182,865 liters of LOX are shipped from the plant to DC3 with 57 shipments, i.e. roughly one shipment every 6 days. The yearly expected flows of the corresponding transportation links are given in Figure 6. The optimal total cost is $366,624.27/year, which includes $200,000/year for installing DC1 and DC3, $65,320.40/year for the transportation cost from the DCs to customers, $77,444.86/year for the transportation cost from the plant to DCs, $10,163.62/year for fixed ordering cost, $10,301.48/year for the cost of working inventory and $3,393.91/year for the cost of safety stocks in the two installed DCs. The major trade-off for this instance is between DC installation costs, transportation costs and inventory costs.

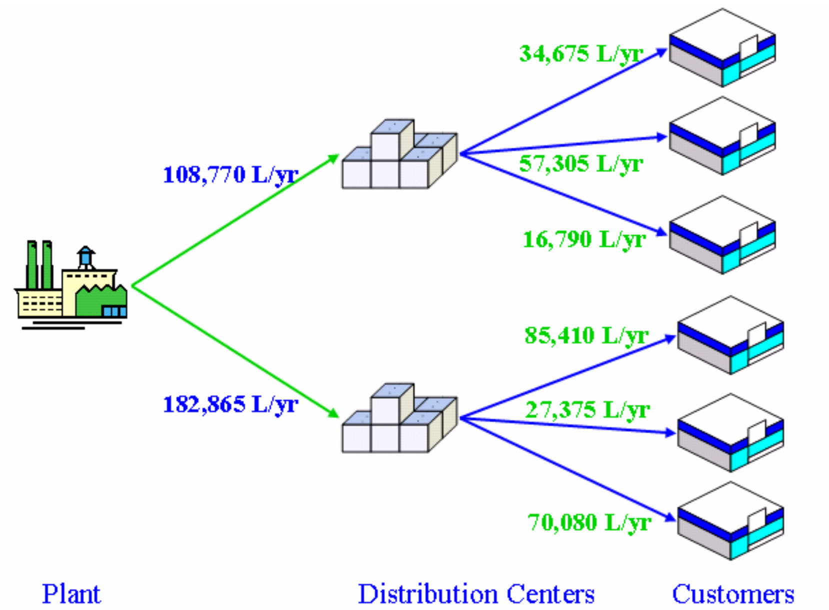


Figure 6 Optimal network structure for the LOX supply chain

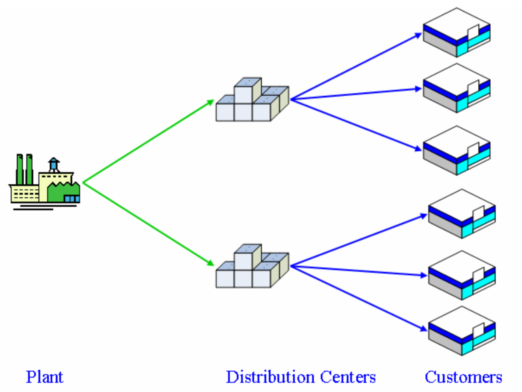
6.2. Illustrative Example for the key trade-offs

To better illustrate the trade-offs in this problem, we consider different weighted parameters for the transportation and inventory cost. All the data for demand uncertainty and transportation costs are the same as the previous example and given in Tables B1-B3 in Appendix B. Other important model coefficients for instances discussed in this section are given in Table B4 in Appendix B. Note that to reveal the trade-offs and using different weighted parameters, the units of some parameters are removed for scaling purpose.

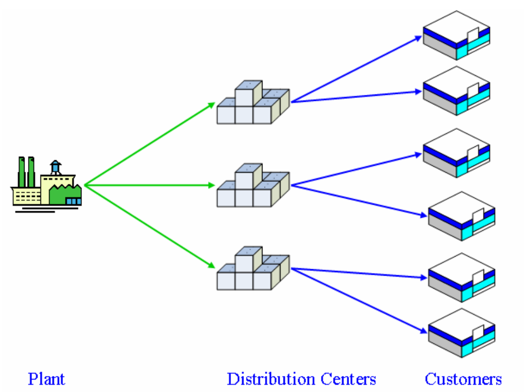
Table 1 Comparison result for the illustrative example

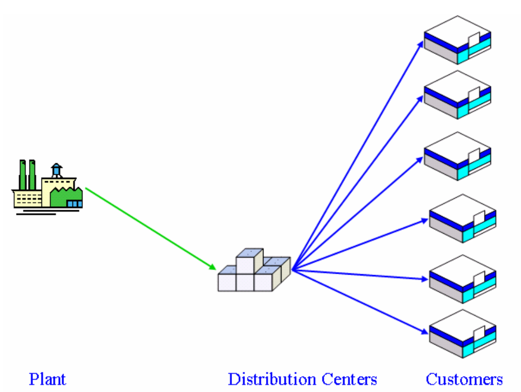
Transportation cost weight factor (β)	Inventory cost weight factor (θ)	Objective Function (*Cost*)	No. DCs	Network structure
0.01	0.01	2260.26	2	Figure 6a
0.1	0.01	8122.93	3	Figure 6b
0.001	0.01	1099.25	1	Figure 6c
0.01	0.1	5359.18	1	Figure 6d
0.01	0.001	1341.04	3	Figure 6e

GAMS/BARON is also used to solve model (**P2**) directly for obtaining global optimal solutions. We first consider a base case with the transportation cost and inventory cost weighted factor as $\beta = 0.01$, $\theta = 0.01$, and then consider different values of weights.

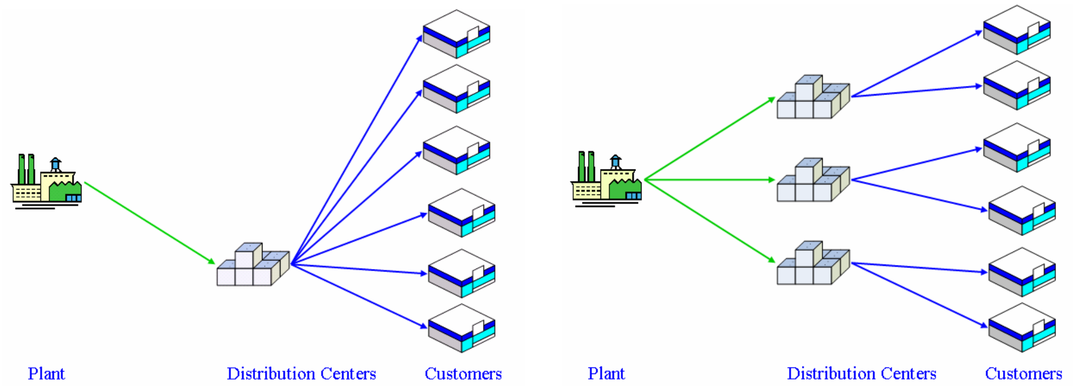


(a) $\beta = 0.01$, $\theta = 0.01$, $Cost = 2260.26$ (base case)





(b) $\beta = 0.1$, $\theta = 0.01$, $Cost = 8122.93$ (c) $\beta = 0.001$, $\theta = 0.01$, $Cost = 1099.25$



(d) $\beta = 0.01$, $\theta = 0.1$, $Cost = 5359.18$ (e) $\beta = 0.01$, $\theta = 0.001$, $Cost = 1341.04$

Figure 7 Optimal LOX supply chain network structure of the illustrative example for different transportation cost and inventory cost weighted parameters

The results for different instances are given in Table 1, and their associated optimal supply chain network structures are given in Figure 7. We can see that in the base case, only two DCs are selected to install and they are connected to three retailers respectively. When we increase the transportation cost factor to $\beta = 0.1$, all the three DCs are installed and each of them serves two retailers (Figure 7b). If we decrease the transportation cost factor to $\beta = 0.001$, only DC 3 is selected to be installed and it serves all the retailers. Thus, the larger the weighted factor for transportation costs β, the more DCs are installed. On the other hand, when we fix the transportation cost factor $\beta = 0.01$, and consider different values of the inventory cost factor θ, we can similarly find out from Figure 7d and 6e that the larger weighted factor for inventory costs, the fewer DCs are installed.

Based on this analysis, we obtain a similar conclusion as Shen et al.[19] that the more DCs are installed, the more transportation costs are potentially reduced, but less inventory cost are saved. The major reason of this performance is that from an inventory cost aspect, the more retailers are pooled to a DC, the more cost saving can be achieved, but from a transportation cost viewpoint, installing more DC to serve different retailers may reduce the total transportation cost. Thus, the trade-off between inventory and transportation costs is established and reflects on the number of DCs besides the tradeoffs for supply chain design costs and operation costs.

7. Computational Results

In order to illustrate the applicability of the proposed solution strategies, we carry out computational experiments for instances with 33, 88 and 150 retailer locations with different weight parameters β and θ. In all cases, each retailer location is also a candidate DC location, i.e. there are as many candidate DC locations as the retailer locations for each instance. Note that in analogy to the industrial gases supply chain introduced in Section 6, the “retailers” correspond to the “customers”.

The model sizes of instancse for n retailer locations (and n candidate DC locations) are given in Table 2. This means that for the largest problem with 150 retailers and 150 candidate distribution centers, the original INLP problem (**P0**) includes 22,650 binary variables and 22,650 constraints, while the reformulated problem (**P2**) includes only 150 binary variables, 22,800 continuous variables and 22,950 constraints.

Table 2 Model statistics of instance for *n* retailer locations (and *n* candidate distribution center locations)

Model	(**P0**)	(**P1**)	(**P2**)	(**P3**)	(**PR**$_j$**(λ)**)
No. of discrete variables	n^2+n	n	n	n	0
No. of continuous variables	0	n^2	n^2+2n	n^2+2n	$n+2$
No. of constraints	n^2+n	n^2+n	n^2+3n	n^2+3n	2

For every retailer i, we set the annual inventory holding cost $h=1$, the service level parameter $z_\alpha=1.96$ (97.5% service level), the order lead time $L=7$ days, the fixed order cost $F_i=10$, the unit shipping cost (from supplier to retailer) $a_i=5$, and the fixed shipping cost (from supplier to retailer) $g_i=10$. Each retailer location represents a city in the U.S., with mean demand μ_i equal to the city population divided by 2000, based on the data from U.S. Census 2000.[54] The standard deviation-to-mean ratio of each customer demand σ_i / μ_i is generated uniformly on $U[0, 0.3]$, and the fixed DC installation cost f_i is generated uniformly on $U[90,110]$.

All the instances are modeled with GAMS[50] and solved with solvers including DICOPT, SBB, Bonmin and BARON on an Intel 3.2 GHz machine with 512 MB RAM.

Tables 3 shows the optimal objective function values for solving problems (**P1**) and (**P2**) directly and with Algorithm 1 by using different solvers, including the outer-approximation algorithm in solver DICOPT, the branch & bound method in solver SBB, and the global optimization solver BARON. We can see that for all the instances we considered, problems (**P1**) and (**P2**) cannot be solved directly by using MINLP solvers such as DICOPT and SBB. This is presumably due to the unbounded gradient when solving the NLP relaxation of (**P1**) and (**P2**). In contrast, with the proposed heuristic algorithm 1 which involves the solution of (**P3**) and (**P2**), we obtain "good quality" solutions by using all the solvers. It is interesting to note that for all the computational instances, the optimal solutions are found in the NLP relaxation step. This shows that the NLP relaxation of the MINLP problems (**P1**) and (**P2**) are quite effective in practice, although theoretically they are not guaranteed to yield integer solutions. Similar conclusions are also reported in Shen et al.,[19] Cornuejols et al.,[34] Conn and Cornuejols.[35]

Table 4 shows the detailed computational times and the objective function values from the computational experiments using different algorithms and solvers to solve instances ranging from 33 to 150 retailer locations with different weight parameters. All the instances are solved with Algorithm 1 by using the global optimization solver BARON and MINLP solvers that rely on convexity assumptions (Bonmin, DICOPT, SBB) with default options, and the proposed Lagrangean heuristic algorithm (Algorithm 2) of which the Lagrangean subproblems (for lower bounds) are solved with the global optimization solver BARON and the feasibility subproblem (for upper bounds) are solved with the NLP solver CONOPT. For comparison purposes, all the instances are also solved with the global optimization solver BARON to obtain global optimal solutions, although BARON failed to terminate the search after more than 10 hours for large instances. By comparing the solution and the associated computational times, we can see that both Algorithms 1 and 2 can obtain good quality solutions that are equal to, or very close to the global optimum.

Algorithm 1 requires much less computational time, and for the smaller instances the solutions obtained are quite close to the global optimum. Note that for all the instances where we can obtain an exact global optimum, the solutions from Algorithm 1 are within 5% of the global optimal solution.

The Lagrangean relaxation algorithm requires longer computational times than Algorithm 1, but the quality of the solutions is significantly improved. The detailed computational results of the Lagrangean relaxation algorithm and the global optimization solver BARON are given in Table 5, where the solutions and computational times are compared. For all instances, the Lagrangean relaxation algorithm requires much shorter computational times than using BARON to obtain the same quality solutions. The instances that BARON can solve to global optimality in less than 10 hours, are solved more efficiently by using the Lagrangean based algorithm in shorter computational times and with 0% optimality gap. For the remaining large scale instances that BARON cannot close the gaps in 10 hours, the optimality gaps of the Lagrangean based algorithm are much smaller (usually less than 1.2%) than the optimality gaps by using BARON for 10 hours.

The results show that good solutions without excessive computational times can be obtained with the proposed Algorithm 1, and near-global solutions can be obtained with the proposed Lagrangean relaxation method (Algorithm 2).

Table 3 Comparison of the optimal objective function values for solving different problems with different solvers and algorithms

No. Retailers	β	θ	**Solving MINLP problem (P1) Directly**		**Solving MINLP problem (P2) Directly**		**Algorithm 1 for MINLP Problem (P2)**			
			DICOPT	SBB	DICOPT	SBB	MILP Relaxation (P3)	SBB	DICOPT	BARON
33	0.001	0.1	Loc. Infeas.	Loc. Infeas.	Loc. Infeas.	Loc. Infeas.	398.64	398.64	398.64	398.64
33	0.001	0.5	Loc. Infeas.	Loc. Infeas.	Loc. Infeas.	Loc. Infeas.	580.46	580.46	580.46	580.46
33	0.005	0.1	Loc. Infeas.	Loc. Infeas.	Loc. Infeas.	Loc. Infeas.	890.44	1023.00	1023.00	1023.00
33	0.005	0.5	Loc. Infeas.	Loc. Infeas.	Loc. Infeas.	Loc. Infeas.	1072.25	1384.03	1384.03	1384.03
88	0.001	0.1	Loc. Infeas.	Loc. Infeas.	Loc. Infeas.	Loc. Infeas.	837.68	935.02	935.02	867.55*
88	0.001	0.5	Loc. Infeas.	Loc. Infeas.	Loc. Infeas.	Loc. Infeas.	1161.12	1386.28	1386.28	1295.02*
88	0.005	0.1	Loc. Infeas.	Loc. Infeas.	Loc. Infeas.	Loc. Infeas.	1956.30	2297.74	2297.74	2297.80*
88	0.005	0.5	Loc. Infeas.	Loc. Infeas.	Loc. Infeas.	Loc. Infeas.	2279.74	3082.19	3082.19	3022.67*
150	0.001	0.5	Loc. Infeas.	Loc. Infeas.	Loc. Infeas.	Loc. Infeas.	1674.08	2205.37	2205.37	1847.93*
150	0.005	0.1	Loc. Infeas.	Loc. Infeas.	Loc. Infeas.	Loc. Infeas.	3107.87	4069.09	4069.09	3689.71*

* Suboptimal solution obtained with BARON for 10 hour limit. Detailed upper and lower bounds are reported in Table 5.

Table 4 Comparison of computational results for solving model (P2) with different algorithms and solvers

No. Retailers	β	θ	**Algorithm 1**								**Algorithm 2**	
			Bonmin (Ipopt)		DICOPT (CONOPT)		SBB (CONOPT)		BARON (**global optimum**)		BARON + CONOPT **Lagrangean**	
			Obj. fun.	Time (s)	Obj. fun.	Time (s)	Obj. fun.	Time (s)	Obj. fun.	Time (s)	Obj. fun.	Time (s)
33	0.001	0.1	398.64	10.125	398.64	0.22	398.64	0.20	398.64	53.31	398.64	15.8
33	0.001	0.2	457.61	366.97	457.61	0.22	457.61	0.25	457.61	54.12	457.61	16.8
33	0.001	0.5	580.46	496.281	580.46	0.23	580.46	0.22	580.46	74.27	580.46	17.9
33	0.001	1.0	728.21	227.828	728.21	0.17	728.21	0.17	728.21	39.14	728.21	15.85
33	0.001	5.0	1460.40	235.765	1460.40	0.18	1460.40	0.20	1460.40	93.22	1460.40	37.28
33	0.001	0.1	398.64	10.125	398.64	0.22	398.64	0.20	398.64	53.31	398.64	15.8
33	0.003	0.1	770.26	274.078	770.26	0.23	770.26	0.27	734.60	75.67	734.60	42.79
33	0.005	0.1	1023.00	244.641	1023.00	0.72	1023.00	0.72	1007.31*	> 10 hr	1006.01	90.85
33	0.008	0.1	1248.59	333.953	1248.59	0.80	1248.59	0.80	1249.37*	> 10 hr	1248.59	53.13
33	0.010	0.1	1418.57	201.610	1418.57	0.80	1418.57	0.76	1398.54*	> 10 hr	1398.39	92.74
88	0.001	0.1	---**	---**	935.02	20.89	935.02	20.94	867.55*	> 10 hr	867.55	356.1
88	0.001	0.5	---**	---**	1386.28	42.16	1386.28	38.50	1295.02*	> 10 hr	1230.99	322.54
88	0.005	0.1	---**	---**	2297.74	42.11	2297.74	42.16	2297.80*	> 10 hr	2284.06	840.28
88	0.005	0.5	---**	---**	3082.19	42.35	3082.19	42.38	3022.67*	> 10 hr	2918.3	934.85
150	0.001	0.5	---**	---**	2205.37	229.57	2205.37	228.31	1847.93*	> 10 hr	1847.93	659.1
150	0.005	0.1	---**	---**	3689.71	397.88	3689.71	398.34	3689.71*	> 10 hr	3689.71	3061.2

* Suboptimal solution obtained with BARON for 10 hour limit. Detailed upper and lower bounds are reported in Table 5.
** Data not available due to solver failure..

Table 5 Comparison of bounds by using the Lagrangean heuristic algorithm and global optimizer BARON

No. Retailers	β	θ	Lagrangean Relaxation (Algorithm 2)					BARON (global optimum)			
			Upper Bound	Lower Bound	Gap	Iterations	Time (s)	Upper Bound	Lower Bound	Optimality Gap	Time (s)
33	0.001	0.1	398.64	398.64	0 %	10	15.8	398.64	398.64	0 %	53.31
33	0.001	0.2	457.61	457.61	0 %	6	16.8	457.61	457.61	0 %	54.12
33	0.001	0.5	580.46	580.46	0 %	6	17.9	580.46	580.46	0 %	74.27
33	0.001	1.0	728.21	728.21	0 %	6	15.85	728.21	728.21	0 %	39.14
33	0.001	5.0	1460.40	1460.40	0 %	13	37.28	1460.40	1460.40	0 %	93.22
33	0.001	0.1	398.64	398.64	0 %	10	15.80	398.64	398.64	0 %	53.31
33	0.003	0.1	734.60	734.60	0 %	16	42.79	734.60	734.60	0 %	75.67
33	0.005	0.1	1006.01	1004.53	0.147 %	32	90.85	1007.31*	965.29	4.353 %	36000
33	0.008	0.1	1248.59	1248.59	0 %	19	53.13	1249.37*	1215.12	2.819 %	36000
33	0.010	0.1	1398.39	1397.7	0.049 %	33	92.74	1398.54*	1364.82	2.471 %	36000
88	0.001	0.1	867.55	867.54	0.001 %	21	356.1	867.55*	837.68	3.566 %	36000
88	0.001	0.5	1230.99	1223.46	0.615 %	24	322.54	1295.02*	1165.15	11.146 %	36000
88	0.005	0.1	2284.06	2280.74	0.146 %	55	840.28	2297.80*	2075.51	10.710 %	36000
88	0.005	0.5	2918.3	2903.38	0.514 %	51	934.85	3022.67*	2417.06	25.056 %	36000
150	0.001	0.5	1847.93	1847.25	0.037 %	13	659.1	1847.93*	1674.08	10.385 %	36000
150	0.005	0.1	3689.71	3648.4	1.132 %	53	3061.2	3689.71*	3290.18	12.143 %	36000

* Suboptimal solution obtained with BARON for 10 hour limit.

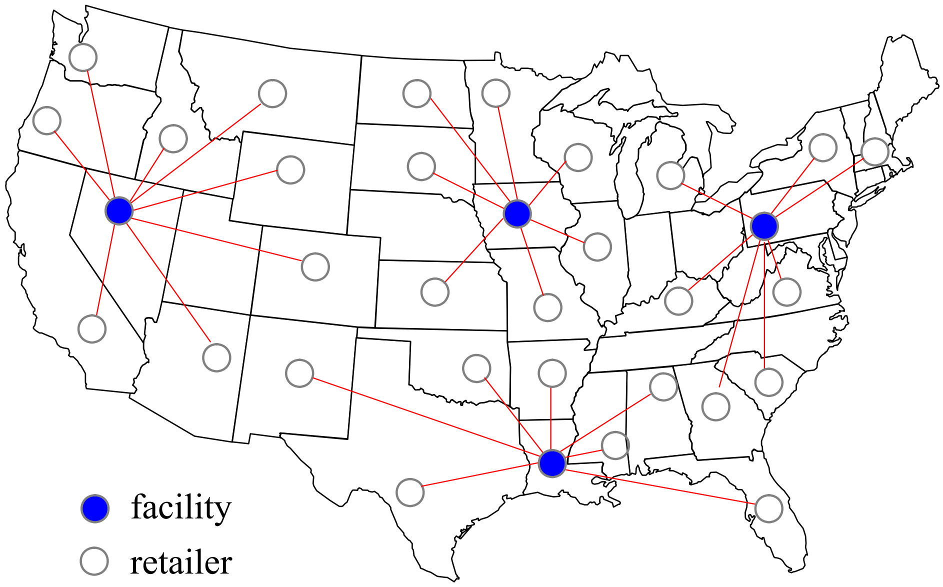
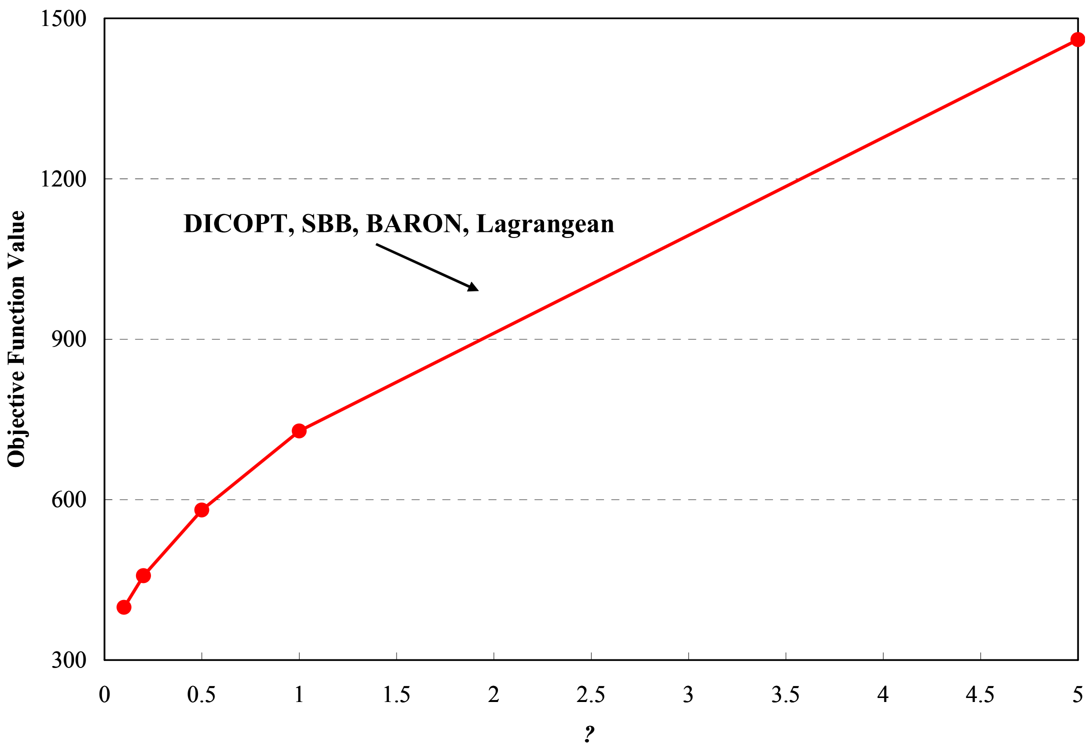
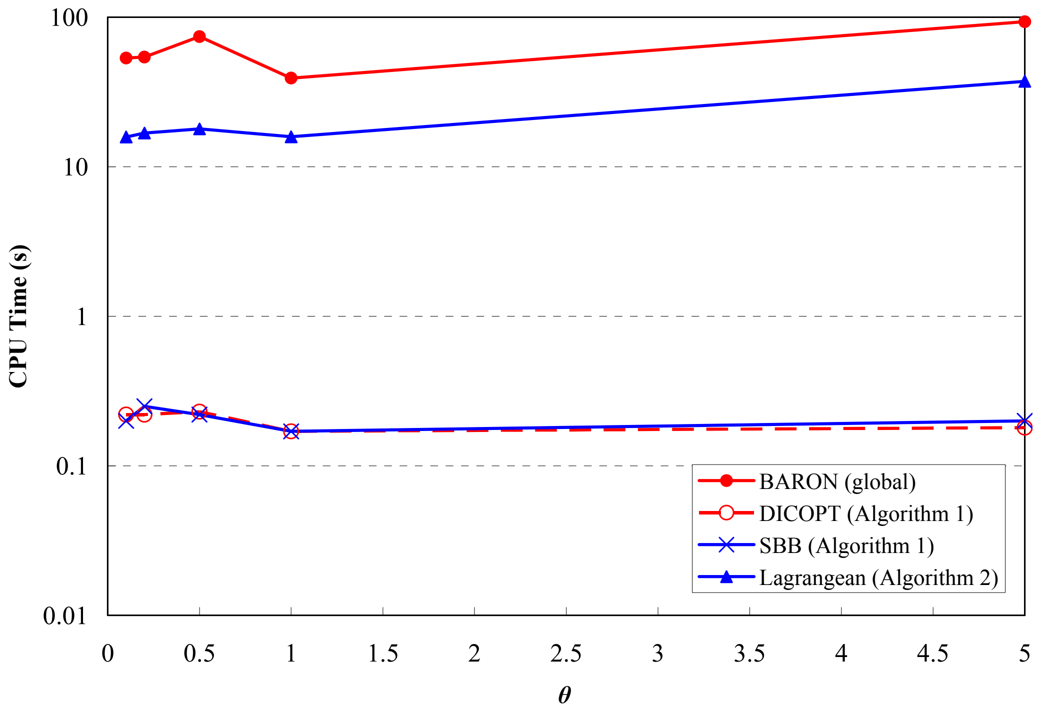


Figure 8 Network structure for the case of 33 retailers with *β*=0.001, *θ*=0.1

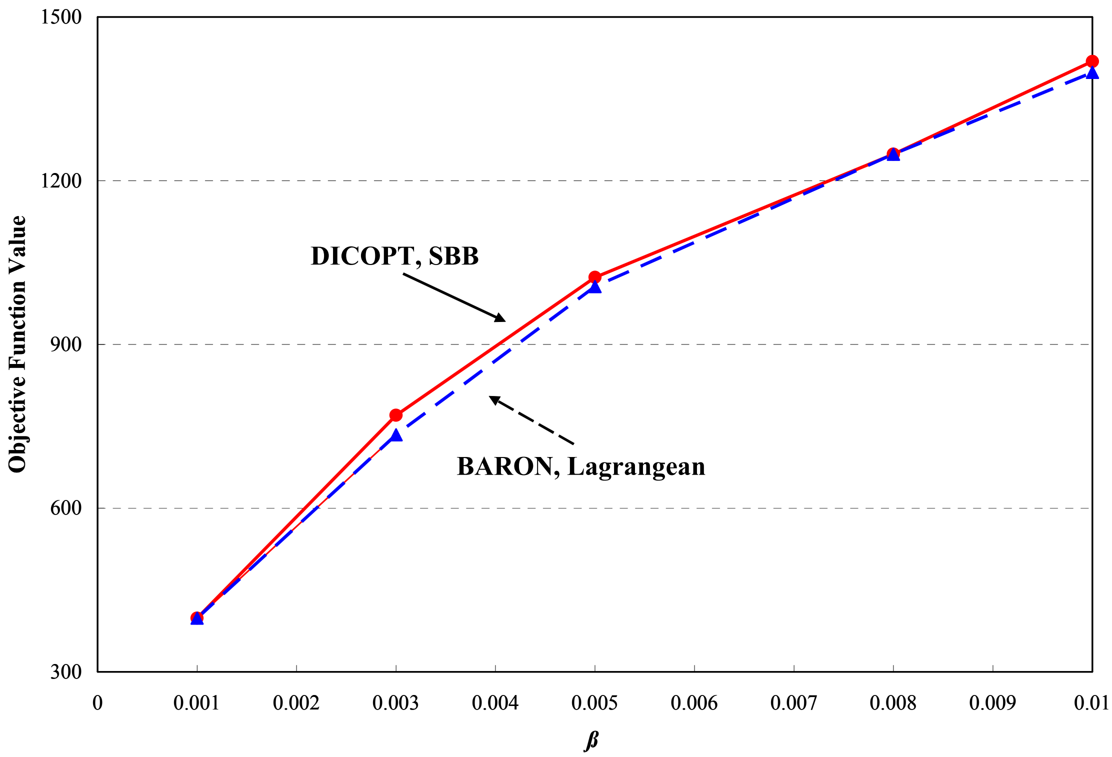


(a) Objective function value

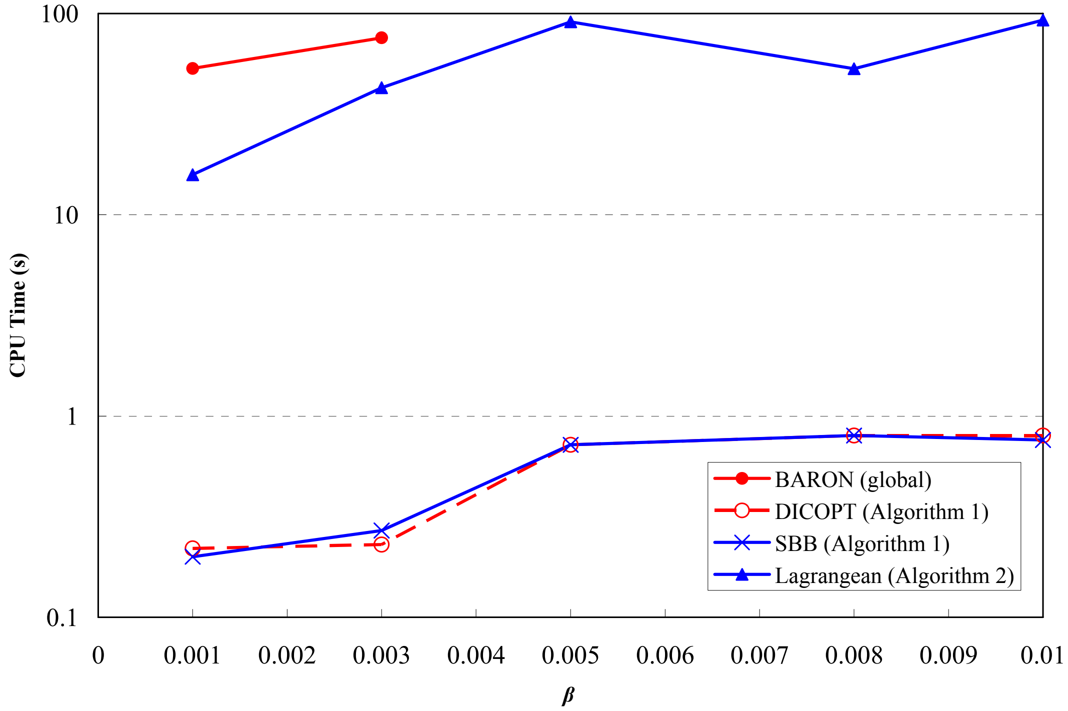


(b) Computational time

Figure 9 Results with different θ values for the case of 33 retailers, β=0.001



(a) Objective function value



(b) Computational time

Figure 10 Results with different β values for the case of 33 retailers, θ=0.1

The network structure of the 33 retailer case with $\beta = 0.001$, $\theta = 0.1$ is given in Figure 8. Figure 9 and Figure 10 show how the objective function values and computational times change as the weights for transportation costs (β) and inventory costs (θ) change. We can see that large weights will lead to an increase of the objective function value for both cases. From Figure 9, we can see that global optimal solutions can be obtained by using either Algorithm 1 or Algorithm 2, but Algorithm 1 requires less computational time. From Figure 10, we can see that although Algorithm 1 with MINLP solvers that rely on convexity assumptions always converges more quickly, the optimal objective function values are often higher. Compared with the global optimizer BARON, the proposed Lagrangean relaxation algorithm can converge to the global optimum in much shorter times for 33 retailers, $\theta = 0.1$ and different β values.

8. Conclusion

This paper has proposed two algorithms for solving the joint supply chain network design and stochastic inventory management model presented by Shen et al.[19] The first algorithm is a heuristic method based on using MINLP optimization methods that rely on assuming convexity in the functions. Computational experiments show that this heuristic algorithm, which includes an initialization scheme, can obtain good quality solutions (typically within 5% of the global optimum). The second algorithm is a heuristic Lagrangean relaxation and decomposition algorithm for obtaining global or near-global optimal solutions. Although there are duality gaps due to the nonconvex nature of the model, extensive numerical examples suggest that the solutions obtained with this algorithm are typically within 1.2% of the global optimum. Moreover, the second algorithm requires much less computational effort than the global optimization solver BARON.

This research can be extended to consider capacity constraints, as both the supplier(s) and the distribution centers are assumed to have infinite capacity in this model. It is likely that Proposition 1 will not hold for the joint capacitated facility location and inventory management models due to the additional capacitated constraints. However, the proposed algorithms can still be used to solve the relaxed problems when branching on the assignment variables.

Another possible extension is to consider different lead times for the distribution centers. In this model the lead times from supplier(s) to all the distribution centers are assumed to be the same, so cost saving can be achieved by risk pooling. If the replenishment lead times of each distribution centers are different, pooling the customers may or may not save costs. It would be also interesting to see how the supply chain network structure and the associated inventory levels change, as the lead time at each distribution center changes, especially when addressing responsiveness issue in the supply chain network design.

Acknowledgment

The authors acknowledge financial support from the Pennsylvania Infrastructure Technology Alliance (PITA) and the National Science Foundation under Grant No. DMI-0556090.

Appendix A: Properties of Model (P0)

In this section, we will present some properties of the INLP model (**P0**). Especially, we show in Proposition 1 that the binary variables for assignment decisions (Y_{ij}) in the model (**P0**) can be relaxed as continuous variables while treating the X_j variables as integer without changing the global optimal integer solution or a local optimal solution for fixed 0-1 values for X_j.

Let us first consider the following relaxation problem (**P1**) with the assignment variables Y_{ij} in (**P0**) relaxed as continuous variables as in constraint (14).

(**P1**) Min: $$\sum_{j\in J}(f_j X_j + \sum_{i\in I}\hat{d}_{ij}Y_{ij} + K_j\sqrt{\sum_{i\in I}\mu_i Y_{ij}} + q\sqrt{\sum_{i\in I}\hat{\sigma}_i^2 Y_{ij}}) \quad (13)$$

s.t. $$\sum_{j\in J} Y_{ij} = 1, \qquad \forall i \in I. \quad (8)$$

$$Y_{ij} \le X_j, \qquad \forall i \in I,\ \forall j \in J. \quad (9)$$

$$X_j \in \{0,1\}, \qquad \forall j \in J \quad (10)$$

$$Y_{ij} \ge 0, \qquad \forall i \in I,\ \forall j \in J. \quad (14)$$

We also consider problem (**P1**) in the reduced space where all the binary variables X_j are fixed to be $X_j^* = 0$ or 1, $\forall j \in J$. We denote this problem as (**AP1**).

(**AP1**) Min: $$\sum_{j\in J}\left(f_j X_j^* + \sum_{i\in I}\hat{d}_{ij}Y_{ij} + K_j\sqrt{\sum_{i\in I}\mu_i Y_{ij}} + q\sqrt{\sum_{i\in I}\hat{\sigma}_i^2 Y_{ij}} \right) \quad (\text{A1})$$

s.t. $$\sum_{j\in J} Y_{ij} = 1, \qquad \forall i \in I. \quad (\text{A2})$$

$$Y_{ij} \le X_j^*, \qquad \forall i \in I,\ \forall j \in J. \quad (\text{A3})$$

$$Y_{ij} \ge 0, \qquad \forall i \in I,\ \forall j \in J \quad (\text{A4})$$

Problem (**P1**) is an MINLP problem, and problem (**AP1**) is a nonlinear programming (NLP) problem with all the binary variables X_j in (**P1**) fixed to certain values. Note that problem (**AP1**) has the following properties given in Lemma 1 and Lemma 2.

Lemma 1. *Problem (AP1) is a concave minimization problem defined over a polyhedron.*

Proof:

It is trivial to see that all the constraints of problem (**AP1**) are linear, therefore the linear constraints correspond to a polyhedron.

Next, we prove that the objective function given in (A1) is concave. Let us assume $\mathbf{Y}^1 = \left\{ Y_{ij}^1 \mid i \in I, j \in J \right\}$ and $\mathbf{Y}^2 = \left\{ Y_{ij}^2 \mid i \in I, j \in J \right\}$ are two feasible solutions satisfying the constraints of problem (**AP1**). Let V^1 and V^2 be the associated objective function values, we have:

$$V^1 = \sum_{j \in J} \left(f_j X_j^* + \sum_{i \in I} \hat{d}_{ij} Y_{ij}^1 + K_j \sqrt{\sum_{i \in I} \mu_i Y_{ij}^1} + q \sqrt{\sum_{i \in I} \hat{\sigma}_i^2 Y_{ij}^1} \right) \tag{A5}$$

$$V^2 = \sum_{j \in J} \left(f_j X_j^* + \sum_{i \in I} \hat{d}_{ij} Y_{ij}^2 + K_j \sqrt{\sum_{i \in I} \mu_i Y_{ij}^2} + q \sqrt{\sum_{i \in I} \hat{\sigma}_i^2 Y_{ij}^2} \right) \tag{A6}$$

Let $0 \le t \le 1$, and $\mathbf{Y}^0 = t \cdot \mathbf{Y}^1 + (1-t) \cdot \mathbf{Y}^2 = \left\{ Y_{ij}^0 = t \cdot Y_{ij}^1 + (1-t) \cdot Y_{ij}^2 \mid i \in I, j \in J \right\}$. Since all the constraints of problem (**AP1**) are linear, it is trivial to show $\mathbf{Y}^0$ is also a feasible solution of problem (**AP1**). Let V^0 be the associated objective function value. We then have,

$$\begin{aligned} V^0 &= \sum_{j \in J} \left(f_j X_j^* + \sum_{i \in I} \hat{d}_{ij} Y_{ij}^0 + K_j \sqrt{\sum_{i \in I} \mu_i Y_{ij}^0} + q \sqrt{\sum_{i \in I} \hat{\sigma}_i^2 Y_{ij}^0} \right) \\ &= \sum_{j \in J} \left(f_j X_j^* + \sum_{i \in I} \hat{d}_{ij} \left[t \cdot Y_{ij}^1 + (1-t) \cdot Y_{ij}^2 \right] + K_j \sqrt{\sum_{i \in I} \mu_i \left[t \cdot Y_{ij}^1 + (1-t) \cdot Y_{ij}^2 \right]} + q \sqrt{\sum_{i \in I} \hat{\sigma}_i^2 \left[t \cdot Y_{ij}^1 + (1-t) \cdot Y_{ij}^2 \right]} \right) \end{aligned} \tag{A7}$$

Thus we have:

$$\begin{aligned} & V^0 - \left[t \cdot V^1 + (1-t) \cdot V^2 \right] \\ &= \sum_{j \in J} K_j \left(\sqrt{t \cdot \sum_{i \in I} \left(\mu_i Y_{ij}^1 \right) + (1-t) \cdot \sum_{i \in I} \left(\mu_i Y_{ij}^2 \right)} - \left[t \cdot \sqrt{\sum_{i \in I} \mu_i Y_{ij}^1} + (1-t) \cdot \sqrt{\sum_{i \in I} \mu_i Y_{ij}^2} \right] \right) \\ &+ \sum_{j \in J} q \left(\sqrt{\left[t \cdot \sum_{i \in I} \left(\hat{\sigma}_i^2 Y_{ij}^1 \right) + (1-t) \cdot \sum_{i \in I} \left(\hat{\sigma}_i^2 Y_{ij}^2 \right) \right]} - \left[t \cdot \sqrt{\sum_{i \in I} \hat{\sigma}_i^2 Y_{ij}^2} + (1-t) \cdot \sqrt{\sum_{i \in I} \hat{\sigma}_i^2 Y_{ij}^2} \right] \right) \end{aligned} \tag{A8}$$

Now consider the following function:

$$f(t) = \sqrt{t \cdot a + (1-t) \cdot b} - \left[t \cdot \sqrt{a} + (1-t) \cdot \sqrt{b} \right] \tag{A9}$$

where $0 \le t \le 1$ and $a \ge 0$, $b \ge 0$. Thus, we have:

$$f(t)=\sqrt{t\cdot a+(1-t)\cdot b}-\left[t\cdot\sqrt{a}+(1-t)\cdot\sqrt{b}\right]=\frac{t\cdot a+(1-t)\cdot b-\left[t\cdot\sqrt{a}+(1-t)\cdot\sqrt{b}\right]^2}{\sqrt{t\cdot a+(1-t)\cdot b}+\left[t\cdot\sqrt{a}+(1-t)\cdot\sqrt{b}\right]}$$

$$=\frac{t\cdot(1-t)\cdot\left(\sqrt{a}-\sqrt{b}\right)^2}{\sqrt{t\cdot a+(1-t)\cdot b}+\left[t\cdot\sqrt{a}+(1-t)\cdot\sqrt{b}\right]}\geq 0 \tag{A10}$$

Comparing (A9), (A10) with each term in (A8), it follows that

$V^0-\left[t\cdot V^1+(1-t)\cdot V^2\right]\geq 0$

Therefore, the objective function of problem (**AP1**) is a concave function. ■

Lemma 2. *The* Y_{ij} *variables take on integer values (0 or 1) for any local or global optimal solution of problem (**AP1**).*

Proof:

Based on Lemma 1, we know that problem (**AP1**) corresponds to the minimization of a concave function over a polyhedron. As proved by Falk and Hoffmann,[55] any local or global optimal solution of this problem always lies on a vertex of the polyhedron.

Furthermore, it is trivial to see the coefficient matrix of constraint (A2) is totally unimodular,[56] while constraints (A3) and (A4) only provide integer bounds of the Y_{ij} variables. Thus, constraints (A2), (A3) and (A4) define an integral polyhedron, of which the extreme points are at the integer values (0 or 1) of the Y_{ij} variables.[56] Therefore, all the Y_{ij} variables equal to 0 or 1 for any local or global optimal solution of problem (**AP1**). ■

Proposition 1. *The continuous variables* Y_{ij} *yield 0-1 integer values when (**P0**) is globally optimized or locally optimized for fixed 0-1 value for* X_j.

Proof:

The MINLP problem (**P1**) is a relaxation of the INLP problem (**P0**) by treating all the Y_{ij} as continuous variables, while problem (**AP1**) is an NLP subproblem of (**P1**) with a fixed value of the integer variables X_j. Based on Lemma 2, we can then conclude that when (**P1**) is globally optimized or locally optimized for fixed 0-1 values for X_j, all the Y_{ij} variables take integer values (0 or 1).

■

Proposition 2. *The global optimal solution of problem (**P2**), or a local optimal solution with fixed 0-1 value for X_j, has all the continuous variables Y_{ij} take on integer value (0 or 1).*

Proof:

Similarly, we consider problem (**P2**) in the reduced space where all the binary variables X_j are fixed to be $X_j^* = 0$ or 1, $\forall j \in J$. We denote this problem as (**AP2**).

(**AP2**) Min: $$\sum_{j\in J}\left(f_j X_j^* + \sum_{i\in I} \hat{d}_{ij} Y_{ij} + K_j Z1_j + qZ2_j \right) \quad \text{(A11)}$$

s.t. $$\sum_{j\in J} Y_{ij} = 1, \qquad \forall i \in I. \quad \text{(A2)}$$

$$Y_{ij} \le X_j^*, \qquad \forall i \in I,\ \forall j \in J. \quad \text{(A3)}$$

$$Y_{ij} \ge 0, \qquad \forall i \in I,\ \forall j \in J \quad \text{(A4)}$$

$$-Z1_j^{\,2} + \sum_{i\in I} \mu_i Y_{ij} \le 0, \qquad \forall j \in J \quad \text{(A12)}$$

$$-Z2_j^{\,2} + \sum_{i\in I} \hat{\sigma}_i^2 Y_{ij} \le 0, \qquad \forall j \in J \quad \text{(A13)}$$

$$Z1_j \ge 0, \qquad \forall j \in J \quad \text{(A14)}$$

$$Z2_j \ge 0, \qquad \forall j \in J \quad \text{(A15)}$$

We associate the Lagrange multiplier λ_i with constraint (A2), μ_{ij} with constraint (A3), t_{ij} with constraint (A4), ρ_j with constraint (A12), γ_j with constraint (A13), ς_j with constraint (A14) and ξ_j with constraint (A15). The Lagrange function for this problem is:

$$L = \sum_{j\in J}\left(f_j X_j^* + \sum_{i\in I} \hat{d}_{ij} Y_{ij} + K_j Z1_j + qZ2_j \right) + \sum_{i\in I} \lambda_i \left(1 - \sum_{j\in J} Y_{ij} \right) + \sum_{i\in I}\sum_{j\in J} \mu_{ij}\left(Y_{ij} - X_j^* \right)$$
$$- \sum_{i\in I}\sum_{j\in J} t_{ij} Y_{ij} + \sum_{j\in J} \rho_j \left(-Z1_j^{\,2} + \sum_{i\in I} \mu_i Y_{ij} \right) + \sum_{j\in J} \gamma_j \left(-Z2_j^{\,2} + \sum_{i\in I} \hat{\sigma}_i^2 Y_{ij} \right) - \sum_{j\in J} \varsigma_j Z1_j - \sum_{j\in J} \xi_j Z2_j$$

Then the KKT conditions for this problem are:

$$\frac{\partial L}{\partial Y_{ij}} = \hat{d}_{ij} - \lambda_i + \mu_{ij} - t_{ij} + \rho_j \mu_i + \gamma_j \hat{\sigma}_i^2 = 0, \forall i \in I, j \in J \tag{A16}$$

$$\frac{\partial L}{\partial Z1_j} = K_j - 2 \cdot \rho_j \cdot Z1_j - \varsigma_j = 0, \ \forall j \in J \tag{A17}$$

$$\frac{\partial L}{\partial Z2_j} = q - 2 \cdot \gamma_j \cdot Z2_j - \xi_j = 0, \ \forall j \in J \tag{A18}$$

$$\mu_{ij}\left(Y_{ij} - X_j^*\right) = 0, \ \forall i \in I, \ \forall j \in J \tag{A19}$$

$$t_{ij} Y_{ij} = 0, \ \forall i \in I, \ \forall j \in J \tag{A20}$$

$$\rho_j \left(-Z1_j^{\ 2} + \sum_{i \in I} \mu_i Y_{ij} \right) = 0, \ \forall j \in J \tag{A21}$$

$$\gamma_j \left(-Z2_j^{\ 2} + \sum_{i \in I} \hat{\sigma}_i^2 Y_{ij} \right) = 0, \ \forall j \in J \tag{A22}$$

$$\varsigma_j \cdot Z1_j = 0, \ \forall j \in J \tag{A23}$$

$$\xi_j \cdot Z2_j = 0, \ \forall j \in J \tag{A24}$$

$$\sum_{j \in J} Y_{ij} = 1, \qquad \forall i \in I. \tag{A2}$$

$$Y_{ij} \le X_j^*, \qquad \forall i \in I, \ \forall j \in J. \tag{A3}$$

$$Y_{ij} \ge 0, \qquad \forall i \in I, \ \forall j \in J \tag{A4}$$

$$-Z1_j^{\ 2} + \sum_{i \in I} \mu_i Y_{ij} \le 0, \qquad \forall j \in J \tag{A12}$$

$$-Z2_j^{\ 2} + \sum_{i \in I} \hat{\sigma}_i^2 Y_{ij} \le 0, \qquad \forall j \in J \tag{A13}$$

$$Z1_j \ge 0, \qquad \forall j \in J \tag{A14}$$

$$Z2_j \ge 0, \qquad \forall j \in J \tag{A15}$$

$$\lambda_i \ge 0, \ \mu_{ij} \ge 0, \ t_{ij} \ge 0, \ \forall i \in I, \ \forall j \in J \tag{A25}$$

$$\rho_j \ge 0, \ \gamma_j \ge 0, \ \varsigma_j \ge 0, \ \xi_j \ge 0, \ \forall j \in J \tag{A26}$$

On the other hand, for problem (**AP1**), if we similarly associate Lagrange multiplier λ_i with constraint (A2), μ_{ij} with constraint (A3) and t_{ij} with constraint (A4). The Lagrange function for problem (**AP1**) is:

$$L' = \sum_{j\in J}\left(f_j X_j^* + \sum_{i\in I}\hat{d}_{ij}Y_{ij} + K_j\sqrt{\sum_{i\in I}\mu_i Y_{ij}} + q\sqrt{\sum_{i\in I}\hat{\sigma}_i^2 Y_{ij}} \right) + \sum_{i\in I}\lambda_i\left(1-\sum_{j\in J}Y_{ij}\right) + \sum_{i\in I}\sum_{j\in J}\mu_{ij}\left(Y_{ij}-X_j^*\right) - \sum_{i\in I}\sum_{j\in J}t_{ij}Y_{ij}$$

Then the KKT conditions for this problem are:

$$\frac{\partial L'}{\partial Y_{ij}} = \hat{d}_{ij} + \frac{K_j\mu_i}{2\sqrt{\sum_{i\in I}\mu_i Y_{ij}}} + \frac{q\hat{\sigma}_i^2}{2\sqrt{\sum_{i\in I}\hat{\sigma}_i^2 Y_{ij}}} - \lambda_i + \mu_{ij} - t_{ij} = 0,\ \forall i\in I, j\in J \qquad \text{(A27)}$$

$$\mu_{ij}\left(Y_{ij}-X_j^*\right) = 0,\ \forall i\in I,\ \forall j\in J \qquad \text{(A19)}$$

$$t_{ij}Y_{ij} = 0,\ \forall i\in I,\ \forall j\in J \qquad \text{(A20)}$$

$$\sum_{j\in J}Y_{ij} = 1, \qquad \forall i\in I. \qquad \text{(A2)}$$

$$Y_{ij} \le X_j^*, \qquad \forall i\in I,\ \forall j\in J. \qquad \text{(A3)}$$

$$Y_{ij} \ge 0, \qquad \forall i\in I,\ \forall j\in J \qquad \text{(A4)}$$

$$\lambda_i \ge 0,\ \mu_{ij}\ge 0,\ t_{ij}\ge 0,\ \forall i\in I,\ \forall j\in J \qquad \text{(A25)}$$

By substituting equations (A17), (A18), (A21), (A22), (A23) and (A24) into (A27), we can have

$$\frac{\partial L'}{\partial Y_{ij}} = \hat{d}_{ij} + \frac{K_j\mu_i}{2\sqrt{\sum_{i\in I}\mu_i Y_{ij}}} + \frac{q\hat{\sigma}_i^2}{2\sqrt{\sum_{i\in I}\hat{\sigma}_i^2 Y_{ij}}} - \lambda_i + \mu_{ij} - t_{ij} = \hat{d}_{ij} - \lambda_i + \mu_{ij} - t_{ij} + \rho_j\mu_i + \gamma_j\hat{\sigma}_i^2 = \frac{\partial L}{\partial Y_{ij}}$$

which is the same as (A14).

This then shows that the KKT conditions of problem (**AP1**) and (**AP2**) are equivalent, although due to the square root terms in (A23) problem (**AP1**) may have unbounded gradients. Because we know that all the optimal solutions of problem (**AP1**) have Y_{ij} take on integer values 0 or 1, we can conclude that Y_{ij} variables also have integer values when (**AP2**) is (locally or globally) optimized. Similarly to Proposition 1, we can conclude that problem (**P2**) has all the Y_{ij} take on integer values when it is l globally optimized or locally optimized for fixed X_j.

■

Appendix B: Data for the Illustrative Examples

Table B1 Parameters for demand uncertainty for the illustrative example

	Mean demand μ_i (Liters/day)	Standard Deviation σ_i (Liters/day)
Customer 1	95	30
Customer 2	157	50
Customer 3	46	25
Customer 4	234	80
Customer 5	75	25
Customer 6	192	80

Table B2 Parameters for unit transportation cost (d_{ij}) between DCs and customers ($/Liter)

	DC 1	DC 2	DC 3
Customer 1	0.04	2.00	2.88
Customer 2	0.08	1.36	1.32
Customer 3	0.36	0.08	1.04
Customer 4	0.88	0.10	0.52
Customer 5	1.52	1.80	0.12
Customer 6	3.36	2.28	0.08

Table B3 Parameters for shipping cost between Plant and DCs

	Fixed shipping cost from plant to DC (g_j) ($/shipment)	Unit shipping cost (a_j) ($/Liter)
DC 1	13	0.24
DC 2	10	0.20
DC 3	14	0.28

Table B4 Model coefficients for the illustrative example in Section 6.2

F_j	Fixed order cost per replenishment	10
f_j	Fixed cost to install a DC (annual)	100
z_α	Service level parameter	1.96
L	Order lead time (days)	7
h	Unit inventory holding cost (annual)	12

χ	Days per year	250

Reference

1. Grossmann, I. E., Enterprise-wide Optimization: A New Frontier in Process Systems Engineering. *AIChE Journal* **2005,** 51, 1846-1857.
2. Chopra, S.; Meindl, P., *Supply Chain Management: Strategy, Planning and Operation*. Prentice Hall: Saddle River, NJ, 2003.
3. Daskin, M. S., *Network and Discrete Location: Models, Algorithms, and Applications*. Wiley: New York, 1995.
4. Owen, S. H.; Daskin, M. S., Strategic facility location: A review. *European Journal of Operations Research* **1998,** 111, 423-447.
5. Zipkin, P. H., *Foundations of Inventory Management*. McGraw-Hill: Boston, MA, 2000.
6. Cachon, G. P.; Fisher, M., Supply Chain Inventory Management and the Value of Shared Information. *Management Science* **2000,** 46, (8), 1032-1048.
7. Kok, A. G.; Graves, S. C., *Supply Chain Management: Design, Coordination and Operation*. Elsevier: 2003.
8. Tsiakis, P.; Shah, N.; Pantelides, C. C., Design of Multi-echelon Supply Chain Networks under Demand Uncertainty. *Industrial & Engineering Chemistry Research* **2001,** 40, 3585-3604.
9. Bok, J.-K.; Grossmann, I. E.; Park, S., Supply Chain Optimization in Continuous Flexible Process Networks. *Industrial & Engineering Chemistry Research* **2000,** 39, (5), 1279-1290.
10. Relvas, S.; Matos, H. A.; A.P.F.D., B.-P.; Fialho, J.; Pinheiro, A. S., Pipeline Scheduling and Inventory Management of a Multiproduct Distribution Oil System. *Industrial & Engineering Chemistry Research* **2006,** 45, 7841-7855.
11. Schulz, E. P.; Diaz, M. S.; Bandoni, J. A., Supply chain optimization of large-scale continuous process. *Computers & Chemical Engineering* **2006,** 29, 1305-1316.
12. Jackson, J. R.; Grossmann, I. E., Temporal decomposition scheme for nonlinear multisite production planning and distribution models. *Industrial & Engineering Chemistry Research* **2003,** 42, (13), 3045-3055.
13. Lim, M.-F.; Karimi, I. A., Resource-Constrained Scheduling of Parallel Production Lines Using Asynchronous Slots. *Industrial & Engineering Chemistry Research* **2003,** 42, 6832-6842.
14. Varma, V. A.; Reklaitis, G. V.; Blau, G. E.; Pekny, J. F., Enterprise-wide modeling & optimization – An overview of emerging research challenges and opportunities. *Computers & Chemical Engineering* **2007,** 31, 692-711.
15. Chen, C.-L.; Lee, W.-C., Multi-objective optimization of multi-echelon supply chain networks with uncertain product demands and prices. *Computers & Chemical Engineering* **2004,** 28, 1131-1144.
16. Gupta, A.; Maranas, C. D., Managing demand uncertainty in supply chain planning.

Computers & Chemical Engineering **2003,** 27, 1219-1227.
17. Gupta, A.; Maranas, C. D.; McDonald, C. M., Mid-term supply chain planning under demand uncertainty: customer demand satisfaction and inventory management. *Computers & Chemical Engineering* **2000,** 24, 2613-2621.
18. Jung, J. Y.; Blau, G.; Pekny, J. F.; Reklaitis, G. V.; Eversdyk, D., Integrated safety stock management for multi-stage supply chains under production capacity constraints. In *Computers & Chemical Engineering*, Submitted: 2007.
19. Shen, Z.-J. M.; Coullard, C.; Daskin, M. S., A joint location-inventory model. *Transportation Science* **2003,** 37, 40-55.
20. Fisher, M. L., The Lagrangean relaxation method for solving integer programming problems. *Management Science* **1981,** 27, (1), 18.
21. Fisher, M. L., An application oriented guide to Lagrangian relaxation. *Interfaces* **1985,** 15, (2), 2-21.
22. Guignard, M.; Kim, S., Lagrangean decomposition: A model yielding stronger Lagrangean bounds. *Mathematical Programming* **1987,** 39, 215-228.
23. Beasley, J. E., Lagrangean heuristics for location problems. *European Journal of Operations Research* **1993,** 65, (3), 383-399.
24. Holmberg, K.; Ling, J., A Lagrangean heuristic for the facility location problem with staircase costs. *European Journal of Operations Research* **1997,** 97, (1), 63-74.
25. Sridharan, R., The capacitated plant location problem. *European Journal of Operations Research* **1993,** 87, 203-213.
26. Pirkul, H.; Jayaraman, V., Production, transportation, and distribution planning in a multi-commodity tri-echelon system. *Transportation Science* **1996,** 30, 291-302.
27. Klose, A., An Lagrangean relax-and-cut approach for two-stage capacitated facility location problems. *Journal of the Operational Research Society* **2000,** 126, 408-421.
28. van den Heever, S. A.; Grossmann, I. E.; Vasantharajan, S., A Lagrangean decomposition heuristic for the design and planning of offshore hydrocarbon field infrastructures with complex economic objectives. *Industrial & Engineering Chemistry Research* **2001,** 40, (13), 2857-2875.
29. Neiro, S. M. S.; Pinto, J. M., A general modeling framework for the operational planning of petroleum supply chains. *Computers & Chemical Engineering* **2004,** 28, 871-896.
30. Axsater, S., Using the Deterministic EOQ Formula in Stochastic Inventory Control. *Management Science* **1996,** 42, (6), 830-834.
31. Zheng, Y. S., On properties of stochastic inventory systems. *Management Science* **1992,** 38, 87-103.
32. Eppen, G., Effects of centralization on expected costs in a multi-echelon newsboy problem. *Management Science* **1979,** 25, (5), 498-501.
33. Daskin, M. S.; Coullard, C.; Shen, Z.-J. M., An inventory-location model: formulation, solution algorithm and computational results. *Annals of Operations Research* **2002,** 110, 83-106.

34. Cornuejols, G.; Nemhauser, G. L.; Wolsey, L. A., The uncapaciated facility location problem. In *Discrete Location Theory*, Mirchandani, P.; Francis, R., Eds. John Wiley and Son Inc.: New York, 1990; pp 119-171.
35. Conn, A. R.; Cornuejols, G., A Projection Method for the Uncapacitated Facility Location Problem. *Mathematical Programming* **1990,** 46, 273-298.
36. Ryoo, H. S.; Sahinidis, N. V., A branch-and-reduce approach to global optimization. *Journal of Global Optimization* **1996,** 8, (2), 107-138.
37. Tawarmalani, M.; Sahinidis, N. V., Global optimization of mixed-integer nonlinear programs: A theoretical and computational study. *Mathematical Programming* **2004,** 99, 563-591.
38. Adjiman, C. S.; Androulakis, I. P.; Floudas, C. A., Global optimization of mixed-integer nonlinear problems. *AIChE Journal* **2000,** 16, (9), 1769-1797.
39. Quesada, I.; Grossmann, I. E., A Global Optimization Algorithm for Linear Fractional and Bilinear Programs. *Journal of Global Optimization* **1995,** 6, 39-76.
40. Smith, E. M. B.; Pantelides, C. C., A symbolic reformulation/spatial branch and bound algorithm for the global optimization of nonconvex MINLPs. *Computers & Chemical Engineering* **1999,** 23, 457-478.
41. Kesavan, P.; Allgor, R. J.; Gatzke, E. P.; Barton, P. I., Outer Approximation Algorithms for Separable Nonconvex Mixed-Integer Nonlinear Programs. *Mathematical Programming* **2004,** 100, (3), 517-535.
42. Sahinidis, N. V., BARON: A general purpose global optimization software package. *Journal of Global Optimization* **1996,** 8, (2), 201-205.
43. Grossmann, I. E., Review of Nonlinear Mixed-Integer and Disjunctive Programming Techniques. *Optimization and Engineering* **2002,** 3, 227-252.
44. Leyffer, S., Integrating SQP and branch and bound for mixed integer nonlinear programming. *Computational Optimization and Applications* **2001,** 18, 295-309.
45. Geoffrion, A. M., Generalized Benders Decomposition. *Journal of Optimization Theory and Applications* **1972,** 10, (4), 237-260.
46. Duran, M. A.; Grossmann, I. E., An outer-approximation algorithm for a class of mixed-integer nonlinear programs. *Mathematical Programming* **1986,** 36, (3), 307.
47. Viswanathan, J.; Grossmann, I. E., A combined penalty-function and outer-approximation method for MINLP optimization. *Computers & Chemical Engineering* **1990,** 14, (7), 769-782.
48. Quesada, I.; Grossmann, I. E., An LP/NLP based branch and bound algorithm for convex MINLP optimization problems. *Computers & Chemical Engineering* **1992,** 16, 937-947.
49. Westerlund, T.; Pettersson, F., A cutting plane method for solving convex MINLP problems. *Computers & Chemical Engineering* **1995,** 19, S131-S136.
50. Brooke, A.; Kendrick, D.; Meeraus, A.; Raman, R., GAMS- A User's Manual. In GAMS Development Corp.: 1998.
51. Bonami, P.; Waechter, A.; Biegler, L. T.; Conn, A. R.; Cornuéjols, G.; Grossmann, I. E.; Laird,

C. D.; Lee, J.; Lodi, A.; Margot, F.; Sawaya, N., An Algorithmic Framework for Convex Mixed Integer Nonlinear Programs. In *Technical Report RC23771*, IBM Thomas J. Watson Research Center: Yorktown Heights, NY, 2005.

52. http:// www.coin-or.org

53. Falk, J. E.; Soland, R. M., An Algorithm for separable nonconvex programming problems. *Management Science* **1969,** 15, 550-569.

54. http://www.census.gov/main/www/cen2000.html

55. Falk, J. E.; Hoffman, K. R., A Successive Underestimation Method for Concave Minimization Problems. *Mathematics of Operations Research* **1976,** 1, (3), 251-259.

56. Nemhauser, G. L.; Wolsey, L. A., *Integer and Combinatorial Optimization*. John Wiley and Son Inc.: New York, NY, 1988.