The Stochastic Location Model with Risk Pooling

Lawrence V. Snyder
Lehigh University

Mark S. Daskin
Northwestern University

Chung-Piaw Teo
National University of Singapore

Report No. 04T-017

The Stochastic Location Model with Risk Pooling

Lawrence V. Snyder *

Department of Industrial and Systems Engineering

Lehigh University

Bethlehem, PA, USA

Mark S. Daskin

Department of Industrial Engineering and Management Sciences

Northwestern University

Evanston, IL, USA

Chung–Piaw Teo

Department of Decision Sciences

National University of Singapore

Singapore

December 22, 2003

Abstract

The Location Model with Risk Pooling (LRMP) seeks to locate distribution centers to minimize the sum of fixed location costs, transportation costs, and inventory costs. The risk-pooling effects of consolidating inventory sites are explicitly handled in the location model. In this paper, we present a stochastic version of the LMRP (SLMRP) that optimizes location, inventory, and allocation decisions under random parameters described by discrete scenarios. The goal is to find solutions that minimize the expected cost of the system across all scenarios. The SLRMP framework can also be used to solve multi-commodity and multi-period problems.

We present a Lagrangian-relaxation–based exact algorithm for the SLMRP. The Lagrangian sub-problem is a non-linear integer program, but it can be solved by a low-order polynomial algorithm. We discuss simple variable-fixing routines that can drastically reduce the size of the problem. We present quantitative and qualitative computational results on problems with up to 150 nodes and 9 scenarios, describing both algorithm performance and solution behavior as key parameters change.

Keywords: supply chain management, facility location, inventory, uncertainty modeling, Lagrangian relaxation

*This research was supportd by NSF Grants DMI-9634750 and DMI-9812915. This support is gratefully acknowledged.

1 Introduction

Supply chain network design decisions are by nature costly and difficult to reverse, and their impact spans a long time horizon. During the time when design decisions are in effect, any of the parameters of the problem—costs, demands, distances, lead times—may change drastically. Traditionally, supply chain optimization models have focused on either the strategic aspects of supply chain design (facility location) or the tactical aspects (inventory management), but not both simultaneously. The strategic models have tended to treat parameters as deterministic (though the body of research on stochastic facility location has been growing steadily in the past few decades), while the tactical models tend to assume that the strategic decisions have already been made. Recently, Shen [32], Shen, Coullard, and Daskin [33], and Daskin, Coullard, and Shen [12] developed a location model with risk pooling (LMRP) that explicitly considers expected inventory costs when making facility location decisions, thus combining strategic and tactical decisions into a single model. The LMRP incorporates stochastic demands that follow a normal distribution. However, even this model, by assuming stationarity of the demand distribution, fails to take into account the changing environment in which the supply chain will operate.

In this paper, we present a stochastic version of the LMRP that explicitly handles parameter uncertainty by allowing parameters to be described by discrete scenarios, each with a specified probability of occurrence. The goal is to choose distribution center (DC) locations, assign retailers to DCs, and set inventory levels at DCs to minimize the total expected systemwide cost. We call this model the stochastic location model with risk pooling (SLMRP).

The SLMRP is a two-stage model, in that strategic decisions (facility location) must be made now, before it is known which scenario will come to pass, while tactical decisions (assignment of retailers to DCs, setting inventory levels) are made in the future, after the uncertainty has been resolved. However, by simply multiplying the results by the number of time periods in the planning horizon, one can think of the SLMRP as a multi-period model in which we make strategic decisions now and then make separate tactical decisions in each time period. Thus our model does not require parameters to be time-stationary; demand means and variances may change over time. If we make decisions now that hedge poorly against the various scenarios, we must live with the consequences as parameters change.

While we describe the SLMRP model using a specific class of supply chain design problems to derive the cost model, the reader should note that the proposed solution method uses only the fact that the objective function is concave in the assignment variables. Hence the method can in fact be used to solve a variety of other supply chain design problems. For example, the SLMRP model can also be used to solve the supply chain network design problem described in the popular textbook by Chopra and Meindl [8]. The company (ALKO Inc.) had over 100 parts in its 1999 line and these products were stored in 5 different DCs operated by ALKO, in 5 different regions. The company wanted to determine whether it should consolidate all or some of its parts in a central warehouse. The parts can be classified in 3 categories in terms of volume of sales, and each part exhibits a different demand mean and standard deviation in each region. The central issue in this supply chain problem is to determine whether the parts should be stored in the field (5 regional DCs), or whether some of the parts should be pooled into the central warehouse. Given the parts assignment, the cost of operating the supply chain network consists of:

- *construction cost* of new warehouse and closure of old DCs;
- *cycle inventory cost* depending on the ordering policy;
- *transportation cost* depending on the location of the DCs and warehouse;
- *safety inventory cost* depending on the desired cycle service level and the variance of the aggregate demand of each part served by the warehouse/DCs.

This cost function can be shown to be a concave function in the part-DC assignment variables. This is thus a concave partitioning problem, which falls into the framework of the SLMRP model proposed in this paper. Each scenario in this case corresponds to a part in the model.

The SLMRP is a versatile model for a variety of different complex supply chain design problems. Our main intention in this paper is to show that we can now solve these problems efficiently, using a standard optimization methodology. The structure of this paper is as follows. In Section 2, we briefly discuss the literature on stochastic facility location models, and we review the LMRP and the solution methods that have been proposed to solve it. In Section 3 we extend this model to the stochastic case, formulating the problem and presenting an exact solution procedure. We also discuss the application of the SLMRP to multi-commodity and multi-period problems. We present computational results in Section 4. In Section 5, we summarize our conclusions and discuss avenues for future research.

2 Literature Review

In this section, we briefly review several approaches to stochastic location modeling. More comprehensive surveys on facility location under uncertainty are contained in [3],

[28], and [29]. [9] and [14] overview both deterministic and stochastic facility location. For a more detailed study of facility location theory, see the texts by Daskin [11], Drezner [15], or Hurter and Martinich [20].

In stochastic location problems, the objective function is usually to minimize expected cost or to maximize expected profit or its expected utility. The random parameters can be either continuous, in which case they are generally assumed to be statistically independent of one another, or described by discrete scenarios, each with a fixed probability. The scenario approach has two main drawbacks. One is that identifying scenarios (let alone assigning probabilities to them) is a daunting and difficult task; indeed, it is the focus of an entire branch of stochastic programming theory. The second problem is that one generally wants to identify a relatively small number of scenarios for computational reasons, but this limits the range of options under which decisions are evaluated. But the scenario approach generally results in more tractable models, and furthermore, it has the advantage of allowing parameters to be statistically dependent, which is generally not possible in the continuous parameter approach. Dependence is often necessary to model reality, since demands are often correlated across time periods or geographical regions, costs are often correlated among suppliers, etc.

Sheppard [34] was one of the first authors to propose a scenario approach to facility location. He suggests selecting facility locations to minimize expected cost, though he does not discuss the issue at length. In any stochastic programming problem, one must determine which decision variables are first-stage and which are second-stage; that is, which variables must be set now and which may be set after the uncertainty has been resolved. In stochastic location modeling, locations are generally first-stage decisions while assignments of customers to facilities are second-stage decisions. (If both decisions occur in the first stage, most problems can be reduced easily to deterministic problems in which uncertain parameters are replaced by their means.)

Weaver and Church [40] and Mirchandani, Oudjit, and Wong [27] present algorithms for a multi-scenario version of the P-median problem (PMP). Their algorithms essentially treat the problem as a deterministic PMP with $|I||S|$ customers instead of $|I|$, where I is the set of customers and S is the set of scenarios. Our algorithm for the SLMRP uses a similar idea, though the more complicated nature of the SLMRP objective function prevents a direct application of the earlier algorithms.

Over the past few decades, the field of stochastic programming (see, e.g., the book by Birge and Louveaux [4]) has become increasingly well developed. The two-stage nature of many facility location problems has made location a popular application of general stochastic programming methods. Louveaux [25] presents stochastic versions of the capacitated PMP and capacitated facility location problem (CFLP) in which demand, production costs, and selling prices are random. The goal is to choose facility locations, determine their capacities, and decide which customers to serve and from which facilities to maximize the expected utility of profit. Louveaux and Peeters [26] present a dual-based heuristic for the CFLP model presented in [25], and Laporte, Louveaux, and van Hamme [24] present an optimal algorithm. França and Luna [18] use Benders decomposition to solve a problem that is a combination of the CFLP and the stochastic transportation problem with random demands. Jornsten and Bjorndal [23] choose where and when to locate facilities over time in order to minimize the expected time-discounted cost; production and distribution costs are random.

Another class of models focuses on minimizing the maximum regret across scenarios. The *regret* of a scenario is the difference (absolute or percentage) between the cost of the chosen solution under a given scenario and the cost of the optimal solution for that scenario. Parameters may be described by discrete scenarios or by a range of possible values. Regret problems tend to be more difficult than expected-value problems because of their minimax structure. Much of the literature on minimax-regret location problems presents polynomial algorithms or analytical properties of restricted problems such as 1-median problems or P-medians on tree networks (see, for example, [7], [38], or [1]). The few papers that discuss minimax-regret location problems on general networks present heuristic approaches to these problems (see, for example, [31], [30], and [10]).

Only a few papers describe stochastic approaches to solving more general supply chain network design problems. For example, Vidal and Goetschalckx [39] discuss the importance of incorporating various types of uncertainty into global supply chain design decisions. Butler, Ammons, and Sokol [6] present a model to design a supply chain for a new product launch; their objective function uses both expected cost and regret objectives.

2.1 The Location Model with Risk Pooling (LMRP)

Shen [32], Shen, Coullard, and Daskin [33], and Daskin, Coullard, and Shen [12] formulate a location model with risk pooling, which we will refer to as the LMRP. Given a set of retailers, the problem is to choose a subset of the retailers to serve as distribution centers (DCs) for the other retailers.[1] (We will use the terms "DC" and "facility" interchangeably.) These DCs order a single product from a single supplier at regular intervals and distribute the product to the retailers. The DCs hold *working inventory*

[1] The set of potential DC locations need not be the same as the set of retailers, but throughout this paper we will assume WLOG that they are equal. If there are retailers that are not potential DC sites, their fixed location costs can be set to ∞, and if there are DC sites that are not retailers, their demand can be set to 0.

representing product that has been ordered from the supplier but not yet requested by the retailers and *safety stock inventory* designed to buffer the system against stockouts during (deterministic) ordering lead times.

Let I be the set of retailers, which face independent normal random demands. The firm pays a fixed location cost for establishing a DC at a retailer, as well as a fixed cost for each order placed at a DC and a holding cost for inventory. There are fixed and variable costs for shipping from the supplier to the DCs and a variable cost for shipping from the DCs to the retailers. We wish to choose DC locations to minimize the sum of all of these costs. The notation is as follows:

Parameters

Demand

μ_i mean daily demand at retailer i, for $i \in I$

σ_i^2 variance of daily demand at retailer i, for $i \in I$

Costs

d_{ij} per-unit cost to ship from a DC located at retailer j to retailer i, for $i, j \in I$

f_j fixed cost per year of locating a DC at retailer j, for $j \in I$

F_j fixed cost per order placed to the supplier by a DC located at retailer j, for $j \in I$

g_j fixed cost per shipment from the supplier to a DC located at retailer j, for $j \in I$

a_j per-unit cost to ship from the supplier to a DC located at retailer j, for $j \in I$

h inventory holding cost per unit per year

Weights

β weight factor associated with transportation cost, $\beta \geq 0$

θ weight factor associated with inventory cost, $\theta \geq 0$

Other Parameters

L_j lead time in days from the supplier to a DC located at retailer j

α desired probability of not stocking out at a DC during a retailer lead-time

z_α standard normal deviate such that $P(z \le z_\alpha) = \alpha$

χ number of working days per year

In the notation above (and the analysis below), the time horizon of the model is assumed to be one year. However, one could easily choose a different time horizon, adjusting the values of f_j, h, and χ accordingly.

The key contribution of the LMRP is that inventory costs are computed endogenously within the model, meaning that location and allocation decisions are made simultaneously with inventory decisions. To define the model, we introduce the following

Decision Variables

$$X_j = \begin{cases} 1, & \text{if we locate a DC at retailer } j \\ 0, & \text{otherwise} \end{cases}$$

$$Y_{ij} = \begin{cases} 1, & \text{if retailer } i \text{ is served by a DC at retailer } j \\ 0, & \text{otherwise} \end{cases}$$

The objective function presented in [33] is

$$\sum_{j\in I} f_j X_j + \beta\chi \sum_{j\in I}\sum_{i\in I} \mu_i(d_{ij}+a_j)Y_{ij} + \sum_{j\in I}\sqrt{2\theta h\chi(F_j+\beta g_j)\sum_{i\in I}\mu_i Y_{ij}} + \theta h z_\alpha \sum_{j\in I}\sqrt{\sum_{i\in I} L_j\sigma_i^2 Y_{ij}}.$$

The first term represents the fixed cost of locating facilities. The second term represents the cost to transport goods from the supplier to the DCs as well as the variable shipment costs from the DCs to the retailers. The third term represents the cost of holding working inventory at the DCs, assuming that each DC follows an economic order quantity (EOQ) policy, as well as the fixed costs of shipping from the supplier to the DCs. Finally, the last term represents the cost of holding safety stock at the DCs to maintain a service level of α.

To simplify the notation, we define

$$\begin{aligned}\hat{d}_{ij} &= \beta\chi\mu_i(d_{ij}+a_j)\\ K_j &= \sqrt{2\theta h\chi(F_j+\beta g_j)}\\ \Theta &= \theta h z_\alpha\end{aligned}$$

We can now formulate the location model with risk pooling (LMRP):

$$\text{(LMRP)}\quad \text{minimize}\quad \sum_{j\in I}\left\{f_jX_j+\sum_{i\in I}\hat{d}_{ij}Y_{ij}+K_j\sqrt{\sum_{i\in I}\mu_iY_{ij}}+\Theta\sqrt{\sum_{i\in I}L_j\sigma_i^2Y_{ij}}\right\} \quad (1)$$

$$\begin{aligned}\text{subject to}\quad \sum_{j\in I}Y_{ij} &= 1 && \forall i\in I && (2)\\ Y_{ij} &\le X_j && \forall i\in I,\forall j\in I && (3)\\ X_j &\in \{0,1\} && \forall j\in I && (4)\\ Y_{ij} &\in \{0,1\} && \forall i\in I,\forall j\in I && (5)\end{aligned}$$

Constraints (2) require each retailer be assigned to exactly one DC. Constraints (3) prohibit a retailer from being assigned to a DC that has not been opened. Constraints (4) and (5) are standard integrality constraints.

Note that if $\theta=0$, problem (LMRP) is identical in form to the classical uncapacitated fixed-charge location problem (UFLP; [2]). Unfortunately, the square-root terms in the objective function make the standard algorithms for the UFLP inapplicable for the problem when $\theta>0$. However, [33] and [12] both use modifications of standard algorithms to solve this problem. Their algorithms depend on the following assumption:

Assumption 1 *The variance-to-mean ratio σ_i^2/μ_i is identical for all retailers. That is, for all $i\in I$, $\sigma_i^2/\mu_i=\gamma$ for some constant $\gamma\ge 0$.*[2]

[2] Or $\mu_i=\sigma_i^2=0$. This is useful for implementing the modeling trick described in footnote 1.

This assumption allows the objective function to be further simplified to

$$\sum_{j\in I}\left\{f_jX_j+\sum_{i\in I}\hat{d}_{ij}Y_{ij}+K_j\sqrt{\sum_{i\in I}\mu_iY_{ij}}+\Theta\sqrt{\sum_{i\in I}L_j\gamma\mu_iY_{ij}}\right\}$$
$$=\sum_{j\in I}\left\{f_jX_j+\sum_{i\in I}\hat{d}_{ij}Y_{ij}+\hat{K}_j\sqrt{\sum_{i\in I}\mu_iY_{ij}}\right\} \tag{6}$$

where

$$\hat{K}_j=K_j+\Theta\sqrt{L_j\gamma}.$$

This revised objective function, with one square-root term instead of two, makes possible an efficient solution procedure. If demands are Poisson, Assumption 1 is satisfied exactly. If not, the assumption may still be satisfied approximately. Another effect of Assumption 1 is that the optimal solution will never open retailer j as a DC but serve demands at j from a different DC, an odd circumstance that can happen if Assumption 1 is not satisfied. (See [33] for an example and [12] for a proof that the situation cannot arise if Assumption 1 holds.) Even with Assumption 1, however, it is possible that a retailer is served from a facility other than its closest. Thus, finding the optimal retailer assignments is considerably more difficult in this case than it is in the classical UFLP.

2.2 Solution Procedures for the LMRP

[33] and [32] presented a set-covering algorithm for the LMRP. Subsequently, [12] presented a Lagrangian-relaxation–based algorithm. The pricing problem for the set-covering algorithm and the sub-problem for the Lagrangian algorithm are both equivalent to the following problem:

$$(\mathrm{SP}_j)\qquad \text{minimize}\qquad \sum_{i\in I}b_iZ_i+\sqrt{\sum_{i\in I}c_iZ_i} \tag{7}$$

$$\text{subject to}\qquad Z_i\in\{0,1\}\qquad \forall i\in I \tag{8}$$

where $c_i > 0$ and b_i is unrestricted. This problem must be solved for each facility $j \in I$ at each iteration of the algorithm. The coefficients b_i and c_i depend on j and on the (LP or Lagrangian) dual variables. (SP$_j$) is a non-linear integer program, but it is shown in [33] that the problem can be solved using the following $O(|I|\log|I|)$ algorithm:

Algorithm 1

Step 1: Partition I into three sets as follows:

$$\begin{aligned} I^+ &= \{i|b_i \geq 0\} \\ I^0 &= \{i|b_i < 0 \text{ and } c_i = 0\} \\ I^- &= \{i|b_i < 0 \text{ and } c_i > 0\} \end{aligned}$$

Step 2: Sort the elements of I^- such that

$$\frac{b_1}{c_1} \leq \frac{b_2}{c_2} \leq \cdots \leq \frac{b_n}{c_n},$$

where $n = |I^-|$.

Step 3: Compute the partial sums

$$S_m = \sum_{i \in I^0} b_i + \sqrt{\sum_{i \in I^0} c_i} + \sum_{\substack{i=1 \\ i \in I^-}}^{m} b_i + \sqrt{\sum_{\substack{i=1 \\ i \in I^-}}^{m} c_i}$$

for $m = 0, \ldots, n$. (Note that the first square-root term will equal 0 by the definition of I^0.)

Step 4: Select the value of m that results in the minimum value of S_m, and set

$$Z_i = \begin{cases} 1, & \text{if } i \in I^0 \\ 1, & \text{if } i \in I^- \text{ and } i \leq m \\ 0, & \text{otherwise} \end{cases}$$

If Assumption 1 does not hold, [33] shows that the problem can still be solved using a general purpose submodular function minimization routine, utilizing $O(|I|^7 \log |I|)$ functional evaluations and arithmetic operations (see [21] and [22] for recent improvements in the running time). Recently, Shu, Teo, and Shen [35] showed that the two-square root pricing problem can be solved in $O(|I|^2 \log |I|)$ time.

3 Stochastic Location Model with Risk Pooling (SLMRP)

The LMRP model discussed in the previous section involves random demands, but estimates of μ_i and σ_i^2 may be inaccurate due to poor forecasts, measurement errors, or changing demand patterns. In this section we present a model that allows the modeler to specify several possible future states, or scenarios. Each scenario dictates the demand and cost information that drives the supply chain model. This allows us to hedge against forecast errors or changes in parameters over time.

We chose to model uncertainty using discrete scenarios (as opposed to continuous probability distributions) for several reasons. The primary reason is that it allows us to model dependence among random parameters. Future demands are likely to be correlated, as are costs. Under the continuous approach, such correlation could be modeled, but in all likelihood the problem would be intractable. Even without dependence, stochastic models with continuous parameters are extremely difficult to solve, and stochastic programming researchers have generally stayed away from them. Computational tractability is our second reason for using scenarios, as the solution techniques previously published for the LMRP can be extended to handle the scenario-based problem. Finally, as we discuss in Section 3.3, the scenario framework can be interpreted in a number of different ways, allowing us to use it to model and solve multi-commodity and multi-period versions of the LMRP.

3.1 Formulation

Suppose now that demand means and variances, distances, and costs are random and are described by scenarios, each with a specified probability of occurrence. Location decisions (X) are scenario-independent: they must be made before it is known which scenario will be realized. Assignment decisions (Y) are scenario-*dependent*, so Y_{ij} becomes Y_{ijs}. Inventory decisions are also scenario-dependent, in that the levels of safety stock change once assignments are made and demand means and variances are known, though there are no explicit inventory variables. Note that there are now two levels of randomness: scenarios determine the means and variances of the demands, but once the scenario has been realized, demands are still random according to the specified normal distribution. Our goal is to choose facility locations to *minimize the expected cost of the system.*

Let S be the set of scenarios, indexed by s. We modify our original notation as follows:

Parameters

Demand

μ_{is} mean daily demand at retailer i in scenario s, for $i \in I,\ s \in S$

σ_{is}^2 variance of daily demand at retailer i in scenario s, for $i \in I,\ s \in S$

Costs

d_{ijs} per-unit cost to ship from a DC located at retailer j to retailer i in scenario s, for $i, j \in I,\ s \in S$

Probabilities

q_s probability that scenario s occurs, for $s \in S$

Variables

Assignment Variables

$$Y_{ijs} = \begin{cases} 1, & \text{if retailer } i \text{ is served by a DC at retailer } j \text{ in scenario } s \\ 0, & \text{otherwise} \end{cases}$$

In fact, any of the costs (a_j, g_j, etc.) and parameters (L_j, h, etc.) other than the fixed facility location cost, f_j, can be scenario-dependent (a_{js}, g_{js}, L_{js}, h_s, etc.); the analysis to follow can be modified in a straightforward way to incorporate such scenario dependencies. For simplicity, however, we will assume that only demand means and variances and DC–retailer transportation costs are scenario-dependent.

We can now formulate the stochastic location model with risk pooling (SLMRP):

$$\text{(SLMRP)} \quad \text{minimize} \quad \sum_{s\in S}\sum_{j\in I} q_s \left\{ f_j X_j + \beta\chi \sum_{i\in I} \mu_{is}(d_{ijs} + a_j) Y_{ijs} + \sqrt{2\theta h\chi(F_j + \beta g_j)\sum_{i\in I}\mu_{is}Y_{ijs}} + \theta h z_\alpha \sqrt{\sum_{i\in I} L_j \sigma_{is}^2 Y_{ijs}} \right\}$$

$$= \sum_{s\in S}\sum_{j\in I} q_s \left\{ f_j X_j + \sum_{i\in I} \hat{d}_{ijs} Y_{ijs} + K_j \sqrt{\sum_{i\in I}\mu_{is}Y_{ijs}} + \Theta \sqrt{\sum_{i\in I} L_j \sigma_{is}^2 Y_{ijs}} \right\} \tag{9}$$

subject to

$$\sum_{j\in I} Y_{ijs} = 1 \quad \forall i \in I, \forall s \in S \tag{10}$$

$$Y_{ijs} \le X_j \quad \forall i \in I, \forall j \in I, \forall s \in S \tag{11}$$

$$X_j \in \{0,1\} \quad \forall j \in I \tag{12}$$

$$Y_{ijs} \in \{0,1\} \quad \forall i \in I, \forall j \in I, \forall s \in S \tag{13}$$

The objective function (9) computes the expected value of the individual-scenario costs given by (1), with subscripts s added to the appropriate parameters and variables. In the last line of the objective function, one additional piece of notation is used:

$$\hat{d}_{ijs} = \beta\chi(d_{ijs} + a_j)\mu_{is}.$$

Constraints (10) require each retailer to be assigned to exactly one DC in each scenario. Constraints (11) prohibit a retailer from being assigned to a given DC in any scenario unless that DC has been opened. Constraints (12) and (13) are standard integrality constraints.

We will make the following assumption, which is the stochastic version of Assumption 1:

Assumption 2 *In each scenario $s \in S$, the variance-to-mean ratio σ_{is}^2/μ_{is} is identical for all retailers. That is, for each $s \in S$, there exists $\gamma_s \geq 0$ such that $\sigma_{is}^2/\mu_{is} = \gamma_s$ for all $i \in I$.*

Note that the variance-to-mean ratio γ_s may differ from scenario to scenario. In the case when this assumption is violated, one can modify the solution procedure described in Section 3.2 using the method proposed in [35] to tackle the two square-root case. For ease of exposition, however, we will focus on situations satisfying Assumption 2. This assumption allows us to rewrite the objective function (9) as follows:

$$\begin{aligned} & \sum_{s\in S}\sum_{j\in I} q_s \left\{ f_j X_j + \sum_{i\in I} \hat{d}_{ijs} Y_{ijs} + K_j \sqrt{\sum_{i\in I} \mu_{is} Y_{ijs}} + \Theta \sqrt{\sum_{i\in I} L_j \gamma_s \mu_{is} Y_{ijs}} \right\} \\ = & \sum_{s\in S}\sum_{j\in I} q_s \left\{ f_j X_j + \sum_{i\in I} \hat{d}_{ijs} Y_{ijs} + \hat{K}_{js} \sqrt{\sum_{i\in I} \mu_{is} Y_{ijs}} \right\} \end{aligned} \tag{14}$$

where

$$\hat{K}_{js} = K_j + \Theta\sqrt{L_j \gamma_s}.$$

Problem (SLMRP) looks like (LMRP) with $|I||S|$ retailers instead of $|I|$ and with the objective function coefficients multiplied by the constant q_s. We will utilize this structure in our solution procedure.

3.2 Solution Procedure

Our solution procedure uses Lagrangian relaxation and is similar to the algorithm in [12], which, in turn, is an extension of the standard Lagrangian-relaxation algorithm for the UFLP. (For a general discussion of Lagrangian relaxation, see [16] and [17]; for its application to the UFLP see [11].)

3.2.1 Lower Bound

We relax constraints (10) with Lagrange multipliers λ_{is} to get the following Lagrangian problem:

$$\text{(SLR)} \quad \max_{\lambda \geq 0} \mathcal{L}_\lambda = \min_{X,Y} \quad \sum_{s\in S}\sum_{j\in I} q_s \left\{ f_j X_j + \sum_{i\in I} \hat{d}_{ijs} Y_{ijs} + \hat{K}_{js} \sqrt{\sum_{i\in I} \mu_{is} Y_{ijs}} \right\} + \sum_{s\in S}\sum_{i\in I} \lambda_{is} \left(1 - \sum_{j\in I} Y_{ijs}\right)$$

$$= \sum_{s\in S}\sum_{j\in I} \left\{ q_s f_j X_j + \sum_{i\in I} (q_s \hat{d}_{ijs} - \lambda_{is}) Y_{ijs} + q_s \hat{K}_{js} \sqrt{\sum_{i\in I} \mu_{is} Y_{ijs}} \right\} + \sum_{s\in S}\sum_{i\in I} \lambda_{is} \tag{15}$$

$$\begin{aligned} \text{subject to} \quad Y_{ijs} &\leq X_j && \forall i \in I, \forall j \in I, \forall s \in S && (16) \\ X_j &\in \{0,1\} && \forall j \in I && (17) \\ Y_{ijs} &\in \{0,1\} && \forall i \in I, \forall j \in I, \forall s \in S && (18) \end{aligned}$$

We can restrict $\lambda \geq 0$ since if $\lambda_{is} < 0$, then $\hat{d}_{ijs} - \lambda_i > 0$ and it is never advantageous to set $Y_{ijs} = 1$ for any j, s; thus if $\lambda_{is} < 0$, a tighter bound can always be attained by setting $\lambda_{is} = 0$.

For fixed values of λ, this problem decomposes by j and s. Imagine we had set $X_j = 1$. To determine which Y_{ijs} variables would be set to 1 for a given scenario s, we solve

$$
\begin{aligned}
(\mathrm{SSP}_{js}) \qquad \text{minimize} \quad \tilde{V}_{js} &= \sum_{i \in I} (q_s \hat{d}_{ijs} - \lambda_{is}) Y_{ijs} + \sqrt{\sum_{i \in I} q_s^2 \hat{K}_{js}^2 \mu_{is} Y_{ijs}} && (19) \\
\text{subject to} \qquad & Y_{ijs} \in \{0,1\} \qquad \forall i \in I && (20)
\end{aligned}
$$

By letting

$$
\begin{aligned}
b_i &= q_s \hat{d}_{ijs} - \lambda_{is} \\
c_i &= q_s^2 \hat{K}_{js}^2 \mu_{is} \\
Z_i &= Y_{ijs},
\end{aligned}
$$

problem (SSP_{js}) becomes equivalent to (SP_j) and, for given j, s, can be solved using the technique described in Section 2.2.

$\tilde{V}_{js}$ is called the *benefit* of facility j in scenario s and represents the contribution to the objective function (15) that would result from having facility j open in scenario s. Of course, if facility j is open in one scenario, it must be open in every scenario. Therefore the overall benefit of opening facility j is equal to the sum of all of the scenario-specific benefits:

$$\tilde{V}_j = \sum_{s \in S} \tilde{V}_{js}.$$

To solve (SLR) for fixed λ, we compute $\tilde{V}_{js}$ for each $j \in I, s \in S$, then compute $\tilde{V}_j$ for each j. Facility j is open in the optimal solution if its benefit plus its fixed cost is negative. The fixed cost to open facility j is equal to $f_j \sum_{s \in S} q_s = f_j$ since the scenario probabilities sum to 1. Therefore, we set $X_j = 1$ if

$$\tilde{V}_j + f_j < 0. \tag{21}$$

If $\tilde{V}_j + f_j \geq 0$ for all $j \in I$, then we set $X_j = 1$ for the j that minimizes $\tilde{V}_j + f_j$ since at least one facility must be open in any feasible solution. We set $Y_{ijs} = 1$ if $X_j = 1$ and $Z_i = 1$ in the optimal solution to (SSP$_{js}$). To solve the overall problem (SLR), we find the optimal values of λ using subgradient optimization (see [16], [17], or [11]). The best value of $\mathcal{L}_\lambda$ found during the Lagrangian process serves as a lower bound on (9).

3.2.2 Upper Bound

Each time we solve (SLR), we use the current solution $(\hat{X}, \hat{Y})$ to obtain a feasible solution to (SLMRP). For each $j \in I$, we open a DC at j if $\hat{X}_j = 1$ in the optimal solution to (SLR). In each scenario s, we assign retailers to facilities as follows. We first loop through all retailers i with $\sum_{j \in I} \hat{Y}_{ijs} \geq 1$ and assign i to the facility j with $\hat{Y}_{ijs} = 1$ that increases the objective function least based on the assignments made so far. Next we loop through retailers with $\sum_{j \in I} \hat{Y}_{ijs} = 0$ and assign each retailer to the open DC that increases the objective function least. In both cases we loop through retailers in decreasing order of mean demand μ_{is}. The resulting solution is feasible for (SLMRP) and its cost provides an upper bound on (9).

If the cost of the solution obtained using this procedure is less than the best upper bound found so far, we apply a retailer re-assignment heuristic to it. This heuristic is similar to that described in [12] and involves re-assigning retailers from their currently assigned facility to a different one in a given scenario if doing so reduces the total cost. This is done for each scenario, since retailers may be assigned to different facilities in different scenarios. If at some point all of the demand assigned to a facility has been removed from the facility, one saves the fixed cost associated with the facility in addition to the other costs.

[12] also describes a facility-exchange heuristic that involves swapping a facility out of the solution in favor of a facility not currently in the solution if doing so reduces the total expected cost; this procedure is similar to Teitz and Bart's procedure for the P-median problem [37]. We did not use this heuristic in our computational tests because it is somewhat computationally expensive and the algorithm performed well without it.

3.2.3 Branch and Bound

If, when the Lagrangian procedure terminates, the best lower bound found is equal to the best upper bound (to within some pre-specified tolerance), we have found the optimal solution to (SLMRP). Otherwise, a branch-and-bound procedure is employed to close the gap, with branching performed on the X_j (location) variables. At each node of the branch-and-bound tree, the facility selected for branching is the unfixed facility with the greatest assigned expected demand; if all facilities in the solution have already been forced open, we branch on an arbitrarily selected unforced facility. The variable is first forced to 0 and then to 1. Branching is done in a depth-first manner. The tree is fathomed at a given node if the lower bound at that node is greater than or equal to the objective value of the best feasible solution found anywhere in the tree to date, or if all facilities have been forced open or closed. In theory, if the overall lower bound is still not equal to the best upper bound found when the branch-and-bound procedure terminates, we must branch on the Y_{ijs} (assignment) variables, but this has never occured in computational testing.

3.2.4 Variable Fixing

Suppose that the Lagrangian procedure terminates at the root node of the branch-and-bound tree with the lower bound strictly less than the upper bound. Let UB be the best upper bound found, let $\tilde{V}_j$ be the facility benefits under a particular set of Lagrange multipliers λ, and let LB be the lower bound (the objective value of (SLR)) under the same λ. Suppose further that $X_j = 0$ in the solution to (SLR) found using λ. If

$$\text{LB} + \tilde{V}_j + f_j > \text{UB} \tag{22}$$

then candidate site j *cannot* be part of the optimal solution, so we can fix $X_j = 0$. To see why this is true, imagine that we chose to branch on X_j. Clearly $\text{LB} + \tilde{V}_j + f_j$ is a valid lower bound for the "$X_j = 1$" node (it would be the first lower bound found if we use λ as the initial multipliers at the new child node), so we would fathom the tree at this new node and never again consider setting $X_j = 1$.

Similarly, suppose $X_j = 1$ in the solution to (SLR) found using λ. If

$$\text{LB} - (\tilde{V}_j + f_j) > \text{UB} \tag{23}$$

then candidate site j *must* be part of the optimal solution, so we can fix $X_j = 1$. Note that in this case, $\tilde{V}_j + f_j < 0$ (otherwise we would have opened j), which is why the left-hand side might exceed UB.

We perform these variable-fixing checks twice after processing has terminated at the root node, once using the optimal multipliers λ and once using the most recent multipliers. This procedure is quite effective in forcing variables open or closed because (SLR) tends to produce very tight lower bounds, making (22) or (23) hold for many facilities j. The time required to perform these checks is negligible.

3.3 Multi-Commodity and Multi-Period Problems

Suppose one wanted to solve the LMRP for multiple commodities simultaneously. Since there are no capacity constraints, one might be tempted to aggregate the products and model them as one. But this strategy falsely assumes that risk-pooling benefits apply across products; that is, that holding inventory of one product protects against stockouts of another. However, the SLMRP framework can be used to model this multi-commodity problem by letting S represent the set of products (instead of scenarios), letting $q_s = 1$ for all $s \in S$, and replacing f_j by $f_j/|S|$ in the objective function (9). The objective function (9) is then interpreted as adding the (product-independent) fixed location costs and the (product-dependent) transportation and inventory costs. Constraints (10) say that each retailer must receive each product from exactly one DC (though a retailer may receive different products from different DCs), and constraints (11) say that no retailer may receive any product from a DC that has not been opened. The solution method for the SLMRP is the same under this multi-commodity interpretation, except that now equation (21) is replaced by

$$\tilde{V}_j + f_j|S| < 0$$

since the q_s no longer sum to 1.

One problem with this interpretation is that it implicitly assumes that each product is ordered separately from the supplier. This is because the inventory terms in the objective function are computed assuming each product is ordered using its optimal EOQ policy. Since the objective function includes fixed ordering and shipment costs (F and g), this model fails to take into account the savings that could result from combining products in a single order. Fortunately, our solution methodology requires only that the replenishment cost is concave in the aggregate demand served. This assumption appears to hold for a variety of multi-product lot-sizing problems, including the case problem mentioned in [8].

Furthermore, this multi-product framework does allow us to model *tooling costs* t_{js} for stocking a given product at a given DC. Since the benefit $\tilde{V}_{js}$ is computed for each facility–product pair (j, s), we can add the tooling cost to $\tilde{V}_{js}$ and only use DC j for product s if the benefit is still negative. Then the overall benefit of facility j is

$$\tilde{V}_j = \sum_{s \in S} \min\{0, \tilde{V}_{js} + t_{js}\}.$$

Tooling costs are often encountered in practice and are generally difficult to model since many supply chain design models do not already have binary variables for DC–product pairs. (See, for example, Geoffrion and Graves [19] or Section 12.4 of Bramel and Simchi-Levi [5].)

The SLMRP framework can also be used to model multi-period problems in which the parameters vary from period to period in a deterministic way. In this case, S represents the set of time periods, and parameter values are specified for each period. Again we set $q_s = 1$ for all s and replace f_j by $f_j/|S|$ for all j. Note that in this multi-period model, facilities are located before period 1, while assignments and inventory policies may change over time. It is not a truly dynamic model in which facilities may be opened or relocated over time. The tooling cost t_{js} could still be used in this model, but it would represent a fixed cost for using a DC in a given time period; it is unlikely that the firm would want to construct a DC but let it remain idle in any period. This similarly makes the tooling cost unnecessary in the standard SLMRP.

4 Computational Results

4.1 Experimental Design

We tested our algorithm for the SLMRP on the 49-node, 88-node, and 150-node data sets described in [11]. The 49-node data set represents the capitals of the lower 48 United States plus Washington, DC; the 88-node data set contains the 49-node data set plus the 50 largest cities in the 1990 U.S. census, minus duplicates; and the 150-node data set contains the 150 largest cities in the 1990 U.S. census.

For each data set, we generated 3-, 5-, and 9-scenario problems. We computed the "base" demand by dividing the population given in [11] by 1000; these base demands were used to compute scenario-specific demands for 9 scenarios following the method described in [13]; in brief, this method involves defining an "attractor" point for each scenario and scaling each retailer's demand based on its distance to the attractor point. The total mean demand is the same in all scenarios for a given problem. The demand variance was set equal to the demand mean in all cases (i.e., $\gamma_s = 1$ for all s). Fixed location costs (f_j) were obtained by dividing the fixed cost in [11] by 10 for the 49-scenario problem and by 100 for the 88-node problem; for the 150-node problem, fixed costs for all retailers were set to 100. Retailer locations for scenario 1 were taken directly from [11] for all three problems; for scenarios 2–9, the latitude and longitude values from scenario 1 were multiplied by a random number drawn uniformly from $U[0.95, 1.05]$. This has the effect of making the distances scenario specific. In all cases, great-circle distances were used.

As mentioned in Section 3.1, the ordering and shipping costs may be scenario-specific; we utilized this feature in our test problems. The fixed ordering and shipping costs (F_{js}

and g_{js}, respectively) were set to 10 and the variable shipping cost (a_{js}) was set to 5 for all retailers in scenario 1. In scenarios 2–9, F_{js} and g_{js} were set to a random number drawn uniformly from $U[7.5, 12.5]$ and a_{js} was set to a random number drawn uniformly from $U[3.75, 6.25]$ (i.e., the costs were perturbed by up to 25% in either direction).

The holding cost h, the lead time L_j, and the days per year χ were set to 1. ($\chi = 1$ may seem unrealistic, but the weights β and θ can serve to translate daily parameters into yearly ones instead of χ.) z_α was set to 1.96 (guaranteeing at least a 97.5% service level). We tested five values of the weights β and θ.

The scenario probabilities for the 9-scenario problems are given in [13]; they are: 0.01, 0.04, 0.15, 0.02, 0.34, 0.14, 0.09, 0.16, 0.05. To obtain the 3-scenario problem, we used the first 3 scenarios and scaled the probabilities so they sum to 1 (the new probabilities are 0.05, 0.2, 0.75); and similarly for the 5-scenario problem (obtaining probabilities 0.018, 0.071, 0.268, 0.036, 0.607).

The parameters used for the Lagrangian relaxation procedure are given in Table 1. For a more detailed description of these parameters, see [11]. The notation $\bar{\mu}$ in the table stands for the average mean demand, taken across all retailers and all scenarios. We terminated the branch-and-bound procedure when the optimality gap was less than 0.1%, or when 2,000 CPU seconds had elapsed.

We coded the algorithm in C++ and performed the computational tests on a Dell Inspiron 7500 notebook computer with a 500 MHz Pentium III processor and 128 MB memory.

Table 1: Parameters for Lagrangian relaxation procedure.

Parameter	Value
Maximum number of iterations at root node	1200
Maximum number of iterations at other nodes	400
Number of non-improving iterations before halving α	12
Initial value of α	2
Minimum value of α	0.00000001
Minimum LB–UB gap	0.1%
Initial value for λ_{is}	$10\bar{\mu} + 10f_i$

4.2 Algorithm Performance

Table 2 describes the algorithm's performance for our computational experiments. The columns are as follows.

 # Ret The number of retailers in the problem.

 # Scen The number of scenarios in the problem.

β The value of β.

θ The value of θ.

Overall LB The lower bound obtained from the branch-and-bound process.

Overall UB The objective value of the best feasible solution found during the branch-and-bound process.

Overall Gap The percentage difference between the overall upper and lower bounds.

Root LB The best lower bound obtained during the Lagrangian process at the root node.

Root UB The objective value of the best feasible solution found during the Lagrangian process at the root node.

Root Gap The percentage difference between the root-node upper and lower bounds.

 # Lag Iter The total number of Lagrangian relaxation iterations performed during the algorithm.

 # BB Nodes The number of branch-and-bound nodes explored during the algorithm.

CPU Time (sec.) The number of CPU seconds that elapsed before the algorithm terminated.

The optimal[3] solution was found (and proven to be optimal) at the root node in 29 out of 45 test problems. For the remaining problems, fewer than 10 branch-and-bound nodes were generally needed, though for a few problems more were necessary. In all but three cases, the optimality gap at the root node was less than 1%, and the root-node gap was always less than 3.1%, indicating that the bound provided by the Lagrangian relaxation process is very tight and that even without branch-and-bound, the Lagrangian procedure can be relied upon to generate a good feasible solution. For the two smaller data sets, the algorithm reached a provably optimal solution within the 2000-second limit in all but one case (in fact, in under two minutes in most cases). The algorithm's performance for the 150-node data set was slightly less impressive, with CPU times occasionally exceeding 2000 seconds and the algorithm terminating without a provably optimal solution. This is not surprising since these problems are quite large—for example, the 9-scenario problem has the equivalent of $|I||S| = 1350$ retailers. In addition, the size of the problem increases the time required at each iteration, hence the number of nodes that can be processed before the time limit is reached decreases as $|I|$ and $|S|$ increase.

The results in Table 2 are summarized in Table 3, in which the Overall Gap and CPU Time fields are averaged over β and θ and reported for each number of retailers

[3] If the optimality gap is less than or equal to 0.1%, we refer to the solution as optimal.

Table 2: Algorithm performance.

# Ret	# Scen	β	θ	Overall LB	Overall UB	Overall Gap	Root LB	Root UB	Root Gap	# Lag Iter	# BB Nodes	CPU Time (sec.)
49	3	0.001	0.1	149,741	149,888	0.10%	149,741	149,888	0.10%	123	1	5.7
49	5	0.001	0.1	151,161	151,302	0.09%	150,925	152,282	0.90%	3186	11	153.2
49	9	0.001	0.1	155,347	155,502	0.10%	154,945	159,655	3.04%	5754	19	486.2
49	3	0.005	0.1	303,327	303,626	0.10%	303,327	303,626	0.10%	44	1	2.9
49	5	0.005	0.1	303,527	303,827	0.10%	303,527	303,827	0.10%	205	1	12.5
49	9	0.005	0.1	316,541	316,847	0.10%	316,541	316,847	0.10%	70	1	10.1
49	3	0.005	0.5	312,658	312,970	0.10%	312,658	312,970	0.10%	76	1	5.4
49	5	0.005	0.5	312,191	312,500	0.10%	312,191	312,500	0.10%	188	1	11.9
49	9	0.005	0.5	325,883	326,205	0.10%	325,883	326,205	0.10%	120	1	15.8
49	3	0.005	1	321,280	321,421	0.04%	321,280	321,421	0.04%	37	1	2.6
49	5	0.005	1	320,041	320,351	0.10%	320,041	320,351	0.10%	87	1	7.9
49	9	0.005	1	334,340	334,666	0.10%	334,340	334,666	0.10%	63	1	8.1
49	3	0.005	20	498,378	498,775	0.08%	498,378	498,775	0.08%	76	1	6.0
49	5	0.005	20	493,292	493,771	0.10%	493,292	493,771	0.10%	104	1	7.4
49	9	0.005	20	512,453	512,953	0.10%	512,453	512,953	0.10%	198	1	20.2
88	3	0.001	0.1	25,318	25,320	0.01%	25,318	25,320	0.01%	979	1	58.6
88	5	0.001	0.1	24,487	24,509	0.09%	24,474	24,509	0.14%	1282	3	109.0
88	9	0.001	0.1	25,644	25,668	0.09%	25,597	26,065	1.83%	3578	13	529.6
88	3	0.005	0.1	55,928	55,983	0.10%	55,928	55,983	0.10%	523	1	36.9
88	5	0.005	0.1	54,067	54,120	0.10%	54,067	54,120	0.10%	729	1	71.9
88	9	0.005	0.1	57,998	58,025	0.05%	57,998	58,025	0.05%	1185	1	194.2
88	3	0.005	0.5	60,671	60,684	0.02%	60,671	60,684	0.02%	195	1	19.0
88	5	0.005	0.5	58,628	58,687	0.10%	58,628	58,687	0.10%	628	1	64.7
88	9	0.005	0.5	62,678	62,740	0.10%	62,497	62,740	0.39%	7299	27	1115.3
88	3	0.005	1	64,796	64,859	0.10%	64,796	64,859	0.10%	129	1	12.5
88	5	0.005	1	62,740	62,802	0.10%	62,740	62,802	0.10%	235	1	30.3
88	9	0.005	1	66,839	66,905	0.10%	66,803	66,986	0.27%	13506	53	> 2000.0
88	3	0.005	20	138,876	138,943	0.05%	138,876	138,943	0.05%	455	1	32.0
88	5	0.005	20	136,717	136,775	0.04%	136,717	136,862	0.11%	1041	3	103.1
88	9	0.005	20	142,272	142,370	0.07%	142,272	142,370	0.07%	328	1	66.8
150	3	0.001	0.1	14,847	14,917	0.47%	14,847	14,917	0.47%	15972	47	> 2000.0
150	5	0.001	0.1	15,141	15,155	0.09%	15,140	15,160	0.13%	8071	30	1319.7
150	9	0.001	0.1	15,794	16,210	2.63%	15,794	16,216	2.67%	5269	14	> 2000.0
150	3	0.005	0.1	23,739	23,763	0.10%	23,739	23,763	0.10%	1105	1	132.4
150	5	0.005	0.1	23,858	23,882	0.10%	23,817	23,882	0.27%	1890	5	340.4
150	9	0.005	0.1	24,137	24,161	0.10%	24,137	24,161	0.10%	1054	1	304.8
150	3	0.005	0.5	32,970	32,992	0.07%	32,969	33,016	0.14%	1223	3	157.7
150	5	0.005	0.5	33,044	33,077	0.10%	32,988	33,080	0.28%	1839	5	319.2
150	9	0.005	0.5	33,691	33,725	0.10%	33,691	33,725	0.10%	994	1	306.5
150	3	0.005	1	40,804	40,843	0.09%	40,804	40,843	0.09%	876	1	124.5
150	5	0.005	1	40,943	40,984	0.10%	40,876	40,987	0.27%	1831	5	309.9
150	9	0.005	1	41,876	41,918	0.10%	41,876	41,918	0.10%	997	1	322.5
150	3	0.005	20	155,653	155,781	0.08%	155,653	155,781	0.08%	227	1	37.0
150	5	0.005	20	157,380	157,508	0.08%	157,134	157,595	0.29%	3029	9	644.9
150	9	0.005	20	161,542	161,812	0.17%	161,369	161,983	0.38%	6107	18	> 2000.0

Table 3: Algorithm performance summary.

# Ret	# Scen	Avg. Overall Gap	Avg. CPU Time
49	3	0.08%	4.5
	5	0.10%	38.6
	9	0.10%	108.1
	Avg	0.09%	50.4
88	3	0.05%	31.8
	5	0.09%	75.8
	9	0.08%	797.9
	Avg	0.07%	301.8
150	3	0.16%	499.6
	5	0.09%	586.8
	9	0.62%	1011.5
	Avg	0.29%	699.3

and scenarios.

4.3 Variable Fixing and DC Locations

Table 4 gives information about the variable-fixing routine and the number of facilities opened. The first four columns are as described above. The other columns are as follows:

 # Fixed Open The number of facilities fixed open by the variable-fixing routine after processing at the root node.

 # Fixed Closed The number of facilities fixed closed by the variable-fixing routine after processing at the root node.

Total # Fixed The sum of the previous two columns.

 # Facil The number of facilities open in the best solution found.

As one might expect, the number of facilities forced into or out of the solution by the variable-fixing routine is larger for problems that have a smaller optimality gap at the root node. (Note that for problems that were solved to optimality at the root node, the variable-fixing routine is unnecessary; we performed it simply for completeness.)

For given values of β and θ, the number of DCs open in the optimal solution does not seem to increase or decrease with any regularity as the number of scenarios increases. However, it is evident that as β increases (from 0.001 to 0.005), the number of DCs increases. This is because when β is large, the transportation term becomes more significant in the objective function, making it desirable to have more DCs. Similarly, as θ increases, the number of DCs decreases because inventory becomes more expensive and risk-pooling becomes more attractive. These trends confirm results reported in [33].

The results in Table 4 are summarized in Table 5, which reports, for each number of retailers and scenarios, the average percentage of facilities fixed open or closed by the variable-fixing routine and the average number of facilities open in the best solution found.

4.4 Stochastic vs. Deterministic Solutions

Table 6 indicates the differences between the stochastic (i.e., min-expected-cost) solutions and the individual scenario solutions. The first four columns are as described above, and the remaining columns are as follows:

 # DCs Different The average, minimum, and maximum (across scenarios) number of DC locations in the stochastic solution that are different from locations in the single-scenario solutions, computed as the number of facilities in the stochastic solution that are not in the scenario solution plus the number of facilities in the scenario solution that are not in the stochastic solution.

% Regret The average, minimum, and maximum (across scenarios) percentage regret that would result from implementing the best stochastic solution found instead of

Table 4: Variable-fixing and DC locations.

# Ret	# Scen	β	θ	# Fixed Open	# Fixed Closed	Total # Fixed	# Facil
49	3	0.001	0.1	6	36	42	10
49	5	0.001	0.1	0	16	16	11
49	9	0.001	0.1	0	4	4	10
49	3	0.005	0.1	15	17	32	30
49	5	0.005	0.1	8	14	22	26
49	9	0.005	0.1	3	8	11	30
49	3	0.005	0.5	12	15	27	30
49	5	0.005	0.5	4	15	19	26
49	9	0.005	0.5	3	9	12	30
49	3	0.005	1	17	17	34	30
49	5	0.005	1	7	15	22	26
49	9	0.005	1	7	8	15	30
49	3	0.005	20	9	18	27	21
49	5	0.005	20	9	19	28	22
49	9	0.005	20	5	10	15	24
88	3	0.001	0.1	9	69	78	13
88	5	0.001	0.1	1	59	60	15
88	9	0.001	0.1	0	14	14	16
88	3	0.005	0.1	14	31	45	44
88	5	0.005	0.1	10	26	36	41
88	9	0.005	0.1	15	30	45	45
88	3	0.005	0.5	36	41	77	41
88	5	0.005	0.5	6	29	35	41
88	9	0.005	0.5	0	7	7	43
88	3	0.005	1	14	29	43	42
88	5	0.005	1	4	28	32	40
88	9	0.005	1	0	9	9	41
88	3	0.005	20	7	61	68	19
88	5	0.005	20	2	55	57	17
88	9	0.005	20	0	40	40	20
150	3	0.001	0.1	0	0	0	53
150	5	0.001	0.1	0	25	25	52
150	9	0.001	0.1	0	0	0	58
150	3	0.005	0.1	50	0	50	133
150	5	0.005	0.1	4	0	4	138
150	9	0.005	0.1	23	0	23	146
150	3	0.005	0.5	16	0	16	123
150	5	0.005	0.5	4	0	4	124
150	9	0.005	0.5	11	0	11	131
150	3	0.005	1	14	4	18	108
150	5	0.005	1	3	0	3	115
150	9	0.005	1	7	0	7	127
150	3	0.005	20	1	0	1	34
150	5	0.005	20	2	0	2	33
150	9	0.005	20	0	0	0	40

Table 5: Variable-fixing and DC locations summary.

# Ret	# Scen	Avg. % Fixed	Avg. # Facil
49	3	66.1%	24.2
	5	43.7%	22.2
	9	23.3%	24.8
	Avg	44.4%	23.7
88	3	70.7%	31.8
	5	50.0%	30.8
	9	26.1%	33.0
	Avg	48.9%	31.9
150	3	11.3%	90.2
	5	5.1%	92.4
	9	5.5%	100.4
	Avg	7.3%	94.3

the optimal solution for a given scenario.

 # Scen-Spec Assign The number of retailers that are assigned to different DCs in different scenarios in the best stochastic solution found.

Clearly, the stochastic solution and the single-scenario solutions differ substantially in their choices of DC locations. This suggests that each of the single-scenario solutions would perform poorly in long-run expected cost. Furthermore, implementing the stochastic solution will entail roughly 8% regret on average and nearly 25% regret in the worst case. Finally, we note that quite a few retailers—roughly half on average, but up to 97%—are assigned to different DCs in different scenarios, indicating the value of allowing retailer assignments to be scenario dependent.

The results in Table 6 are summarized in Table 7, which reports, for each number of retailers and scenarios, the averages across β and θ of the "Avg # DCs Different", "Avg % Regret", and "# Scen-Spec Assign" columns.

Table 6: Stochastic vs. deterministic solutions.

# Ret	# Scen	β	θ	# DCs Different Avg	Min	Max	% Regret Avg	Min	Max	# Scen-Spec Assign
49	3	0.001	0.1	8.3	1	13	6.5%	0.5%	12.1%	29
49	5	0.001	0.1	12.4	8	16	10.4%	3.9%	21.2%	36
49	9	0.001	0.1	9.0	5	13	7.2%	4.1%	11.1%	36
49	3	0.005	0.1	7.3	4	9	5.7%	0.6%	8.2%	18
49	5	0.005	0.1	11.2	2	16	10.2%	0.9%	16.6%	23
49	9	0.005	0.1	10.6	6	14	9.5%	4.9%	18.9%	19
49	3	0.005	0.5	9.0	4	13	5.5%	0.7%	8.1%	18
49	5	0.005	0.5	11.2	3	16	9.8%	0.9%	16.1%	23
49	9	0.005	0.5	10.7	7	13	9.3%	4.9%	18.4%	19
49	3	0.005	1	10.0	7	13	5.4%	0.8%	8.1%	18
49	5	0.005	1	11.2	3	15	9.5%	0.9%	15.6%	23
49	9	0.005	1	11.7	7	16	9.2%	4.9%	18.0%	19
49	3	0.005	20	10.0	1	16	8.4%	0.1%	14.3%	28
49	5	0.005	20	12.2	8	16	7.1%	2.3%	12.3%	29
49	9	0.005	20	12.4	8	20	7.1%	3.6%	11.8%	32
88	3	0.001	0.1	10.0	2	15	7.7%	0.1%	12.4%	48
88	5	0.001	0.1	14.0	8	20	8.5%	2.2%	12.6%	61
88	9	0.001	0.1	13.1	8	16	7.6%	2.6%	12.3%	70
88	3	0.005	0.1	16.0	4	26	8.5%	0.6%	15.4%	36
88	5	0.005	0.1	22.2	11	25	9.9%	1.8%	15.5%	47
88	9	0.005	0.1	22.8	16	29	11.1%	5.6%	17.9%	43
88	3	0.005	0.5	16.7	5	28	8.5%	0.5%	17.2%	38
88	5	0.005	0.5	22.2	12	28	9.0%	1.9%	14.4%	47
88	9	0.005	0.5	23.2	18	30	10.5%	6.2%	17.5%	46
88	3	0.005	1	18.0	10	24	8.3%	0.5%	16.6%	37
88	5	0.005	1	22.2	11	26	8.8%	2.0%	15.5%	49
88	9	0.005	1	23.0	16	33	10.5%	5.2%	19.9%	51
88	3	0.005	20	12.7	5	18	3.4%	0.4%	5.9%	56
88	5	0.005	20	14.6	7	20	5.4%	0.7%	8.1%	65
88	9	0.005	20	16.7	12	23	5.8%	2.7%	9.8%	75
150	3	0.001	0.1	37.0	14	54	10.6%	1.3%	20.7%	94
150	5	0.001	0.1	42.4	21	53	10.4%	1.0%	20.8%	105
150	9	0.001	0.1	43.7	33	51	13.2%	5.7%	23.7%	109
150	3	0.005	0.1	33.7	10	64	10.2%	0.8%	24.4%	31
150	5	0.005	0.1	28.0	17	55	8.2%	2.1%	22.2%	26
150	9	0.005	0.1	24.9	16	55	5.6%	2.8%	19.9%	23
150	3	0.005	0.5	38.3	14	63	9.2%	0.8%	20.2%	60
150	5	0.005	0.5	35.0	24	56	7.5%	1.8%	19.0%	66
150	9	0.005	0.5	32.9	23	57	6.7%	2.8%	18.8%	73
150	3	0.005	1	43.3	21	65	8.8%	0.6%	19.6%	73
150	5	0.005	1	38.6	21	58	6.7%	1.6%	14.6%	75
150	9	0.005	1	38.6	29	58	6.5%	3.0%	16.1%	83
150	3	0.005	20	25.7	13	40	1.6%	0.8%	3.0%	130
150	5	0.005	20	29.4	16	39	2.8%	0.7%	5.1%	134
150	9	0.005	20	34.9	25	48	2.8%	1.2%	5.4%	146

Table 7: Stochastic vs. deterministic solutions summary.

# Ret	# Scen	Avg # DCs Different	Avg % Regret	Avg # Scen-Spec Assign
49	3	8.9	6.3%	22.2
	5	11.6	9.4%	26.8
	9	10.9	8.5%	25.0
	Avg	10.5	8.0%	24.7
88	3	14.7	7.3%	43.0
	5	19.0	8.3%	53.8
	9	19.8	9.1%	57.0
	Avg	17.8	8.2%	51.3
150	3	35.6	8.1%	77.6
	5	34.7	7.1%	81.2
	9	35.0	7.0%	86.8
	Avg	35.1	7.4%	81.9

5 Conclusions

In this paper we introduced the stochastic location model with risk pooling (SLMRP), which simultaneously optimizes location, assignment, and inventory in a supply chain under scenario-based parameters. We formulated the problem as a non-linear integer program and presented a Lagrangian relaxation algorithm for it. The Lagrangian subproblem is also a non-linear integer progrem, but we showed that using a previously published algorithm, this problem can be solved in $O(|I|\log|I|)$ time for each facility–scenario pair. We described how to use the SLMRP framework to model multi-commodity and multi-period problems. Our computational results show the algorithm to be quite effective. Tests on problems with up to 150 facilities and 9 scenarios generally resulted in solution times less than two minutes, root-node optimality gaps of less than 1%, and only a few branch-and-bound nodes needed.

We have identified three main avenues for future research on the SLMRP. The first is to solve the SLMRP when DC–retailer assignments are scenario independent. The general purpose submodular function minimization routine could be used to solve the pricing problem in this case, but it would have utilize $O(|I|^7 \log |I|)$ function evaluations and arithmetic operatics for each facility j at each Lagrangian iteration. Unfortunately, the method in [35] for the two sqaure-root case would have complexity $O(|I|^{|S|} \log |I|)$ if extended to the multiple-square-root problem, making it efficient only when the number of scenarios is small. Further research should be undertaken to solve this problem, as some firms perfer to keep DC–retailer assignments fixed over time. Scenario-independent assignments are also important in the multi-commodity interpretation of the SLMRP, representing single-sourcing restrictions imposed by many firms.

It is evident from Table 6 that the optimal stochastic solution may perform well in the expected value but poorly in certain scenarios. This may be unacceptable for some decision makers, especially those whose decisions are evaluated *ex post* and who may have only one chance "get it right." We have already begun to study models that restrict or minimize the regret in each scenario while also maintaining a low expected cost (see [36]). Although computationally tractable, these models are significantly more difficult to solve than is the SLMRP.

Finally, as mentioned in Section 3.3, the problem with modeling multi-commodity problems under the SLMRP framework is that inventory and supplier–DC transportation costs are computed as though products are each ordered from the supplier separately. This does not accurately model the situation faced by most firms. We have begun to undertake further research to determine how to address this issue and, more generally, which inventory systems can be modeled using the SLMRP framework.

References

[1] I. Averbakh, O. Berman, Minmax regret median location on a network under uncertainty, INFORMS Journal on Computing 12 (2) (2000) 104-110.

[2] M.L. Balinski, Integer programming: Methods, uses, computation, Management Science 12 (3) (1965) 253-313.

[3] O. Berman, D. Krass, Facility location problems with stochastic demands and congestion, in: Z. Drezner, H.W. Hamacher (Eds.), Facility Location: Applications and Theory, Springer-Verlag, New York, 2001, pp. 331-373.

[4] J.R. Birge, F. Louveaux, Introduction to Stochastic Programming, Springer, New York, 1997.

[5] J. Bramel, D. Simchi-Levi, The Logic of Logistics: Theory, Algorithms, and Applications for Logistics Management, Springer, New York, 1997.

[6] R.J. Butler, J.C. Ammons, J. Sokol, A robust optimization model for strategic production and distribution planning for a new product, 2003, Working Paper.

[7] B.T. Chen, C.S. Lin, Minmax-regret robust 1-median location on a tree, Networks 31 (2) (1998) 93-103.

[8] S. Chopra, P. Meindl, Supply Shain Management: Strategy, Planning, and Operation, Prentice Hall, Upper Saddle River, NJ, 2001.

[9] J. Current, M.S. Daskin, D. Schilling, Discrete network location models, in: Z. Drezner, H.W. Hamacher (Eds.), Facility Location: Applications and Theory, Springer-Verlag, New York, 2001, pp. 83-120.

[10] J. Current, S. Ratick, C. ReVelle, Dynamic facility location when the total number of facilities is uncertain: A decision analysis approach, European Journal of Operational Research 110 (3) (1997) 597-609.

[11] M.S. Daskin, Network and Discrete Location: Models, Algorithms, and Applications, Wiley, New York, 1995.

[12] M.S. Daskin, C.R. Coullard, Z.-J.M. Shen, An inventory-location model: Formulation, solution algorithm and computational results, Annals of Operations Research 110 (2002) 83-106.

[13] M.S. Daskin, S.M. Hesse, C.S. ReVelle, α-reliable p-minimax regret: A new model for strategic facility location modeling, Location Science 5 (4) (1997) 227-246.

[14] M.S. Daskin, S.H. Owen, Location models in transportation, in: R.W. Hall (Ed.), Handbook of Transportation Science, Kluwer Academic Publishers, Boston, 1999, pp. 311-360.

[15] Z. Drezner, Facility Location: A Survey of Applications and Methods, Springer, New York, 1995.

[16] M.L. Fisher, The Lagrangian relaxation method for solving integer programming problems, Management Science 27 (1) (1981) 1-18.

[17] M.L. Fisher, An applications oriented guide to Lagrangian relaxation, Interfaces 15 (2) (1985) 10-21.

[18] P.M. França, H.P.L. Luna, Solving stochastic transportation-location problems by generalized Benders decomposition, Transportation Science 16 (2) (1982) 113-126.

[19] A.M. Geoffrion, G.W. Graves, Multicommodity distribution system design by Benders decomposition, Management Science 20 (5) (1974) 822-844.

[20] A.P. Hurter, J.S. Martinich, Facility Location and the Theory of Production, Kluwer Academic Publishers, Boston, 1989.

[21] S. Iwata, A faster scaling algorithm for minimizing submodular functions, in: W.J. Cook, A.S. Schulz (Eds.), proceedings of the Integer Programming and Combinatorial Optimization Conference, Cambridge, MA, 2002, Springer, pp. 1-8.

[22] S. Iwata, L. Fleischer, S. Fujishige, A combinatorial strongly polynomial algorithm for minimizing submodular functions, Journal of the ACM 48 (4) (2001) 761-777.

[23] K. Jornsten, M. Bjorndal, Dynamic location under uncertainty, Studies in Regional and Urban Planning 3 (1994) 163-184.

[24] G. Laporte, F.V. Louveaux, L. van Hamme, Exact solution to a location problem with stochastic demands, Transportation Science 28 (2) (1994) 95-103.

[25] F.V. Louveaux, Discrete stochastic location models, Annals of Operations Research 6 (1986) 23-34.

[26] F.V. Louveaux, D. Peeters, A dual-based procedure for stochastic facility location, Operations Research 40 (3) (1992) 564-573.

[27] P.B. Mirchandani, A. Oudjit, R.T. Wong, Multidimensional extensions and a nested dual approach for the m-median problem, European Journal of Operational Research 21 (1) (1985) 121-137.

[28] S.H. Owen, M.S. Daskin, Strategic facility location: A review, European Journal of Operational Research 111 (3) (1998) 423-447.

[29] C. ReVelle, J.C. Williams, Reserve design and facility siting, in: Z. Drezner, H.W. Hamacher (Eds.), Facility Location: Applications and Theory, Springer-Verlag, New York, 2001, pp. 310-330.

[30] D. Serra, V. Marianov, The p-median problem in a changing network: The case of barcelona, Location Science 6 (1-4) (1998) 383-394.

[31] D. Serra, S. Ratick, C. ReVelle, The maximum capture problem with uncertainty, Environment and Planning B 23 (1996) 49-59.

[32] Z.-J.M. Shen, Efficient algorithms for various supply chain problems, Ph.D. Thesis, Department of Industrial Engineering and Management Sciences, Northwestern University, Evanston, IL, 2000.

[33] Z.-J.M. Shen, C.R. Coullard, M.S. Daskin, A joint location-inventory model, Transportation Science 37 (1) (2003) 40-55.

[34] E.S. Sheppard, A conceptual framework for dynamic location-allocation analysis, Environment and Planning A 6 (1974) 547-564.

[35] J. Shu, C.-P. Teo, Z.-J.M. Shen, Stochastic transportation-inventory network design problem, 2001, to appear in Operations Research.

[36] L.V. Snyder, Supply chain robustness and reliability: Models and algorithms, Ph.D. Dissertation, Department of Industrial Engineering and Management Sciences, Northwestern University, Evanston, IL, 2003.

[37] M.B. Teitz, P. Bart, Heuristic methods for estimating generalized vertex median of a weighted graph, Operations Research 16 (5) (1968) 955-961.

[38] G.L. Vairaktarakis, P. Kouvelis, Incorporation dynamic aspects and uncertainty in 1-median location problems, Naval Research Logistics 46 (2) (1999) 147-168.

[39] C.J. Vidal, M. Goetschalckx, Modeling the effect of uncertainties on global logistics systems, Journal of Business Logistics 21 (1) (2000) 95-120.

[40] J.R. Weaver, R.L. Church, Computational procedures for location problems on stochastic networks, Transportation Science 17 (2) (1983) 168-180.