TRANSPORTATION SCIENCE
Vol. 39, No. 3, August 2005, pp. 400–416
ISSN 0041-1655 | EISSN 1526-5447 | 05 | 3903 | 0400

DOI 10.1287/trsc.1040.0107

Reliability Models for Facility Location: The Expected Failure Cost Case

Lawrence V. Snyder
Department of Industrial and Systems Engineering, Lehigh University, Bethlehem, Pennsylvania 18015, lvs2@lehigh.edu

Mark S. Daskin
Department of Industrial Engineering and Management Sciences, Northwestern University, Evanston, Illinois 60208, m-daskin@northwestern.edu

Classical facility location models like the P-median problem (PMP) and the uncapacitated fixed-charge location problem (UFLP) implicitly assume that, once constructed, the facilities chosen will always operate as planned. In reality, however, facilities "fail" from time to time due to poor weather, labor actions, changes of ownership, or other factors. Such failures may lead to excessive transportation costs as customers must be served from facilities much farther than their regularly assigned facilities. In this paper, we present models for choosing facility locations to minimize cost, while also taking into account the expected transportation cost after failures of facilities. The goal is to choose facility locations that are both inexpensive under traditional objective functions and also *reliable*. This reliability approach is new in the facility location literature. We formulate reliability models based on both the PMP and the UFLP and present an optimal Lagrangian relaxation algorithm to solve them. We discuss how to use these models to generate a trade-off curve between the day-to-day operating cost and the expected cost, taking failures into account, and we use these trade-off curves to demonstrate empirically that substantial improvements in reliability are often possible with minimal increases in operating cost.

Key words: facility location; disruptions; Lagrangian relaxation; multiobjective optimization
History: Received: August 2003; revision received: April 2004; accepted: June 2004.

1. Introduction

The uncapacitated fixed-charge location problem (UFLP) is a classical facility location problem that chooses facility locations and assignments of customers to facilities to minimize the sum of fixed and transportation costs. Once a set of facilities has been constructed, however, one or more of them may from time to time become unavailable—for example, due to inclement weather, labor actions, sabotage, or changes in ownership. These facility "failures" may result in excessive transportation costs as customers previously served by these facilities must now be served by more distant ones. The models presented in this paper choose facility locations to minimize day-to-day construction and transportation costs, while also hedging against failures within the system. We call the ability of a system to perform well even when parts of the system have failed the "reliability" of the system. Our goal, then, is to choose facility locations that are both inexpensive and reliable.

Consider the 49-node dataset described in Daskin (1995), consisting of the capitals of the continental United States plus Washington, D.C. Demands are proportional to the 1990 state populations. The optimal UFLP solution for this problem is pictured in Figure 1; this solution entails a fixed cost of $348,000 and a transportation cost of $509,000. (Transportation costs are taken to be $0.00001 per mile per unit of demand.) However, if the facility in Sacramento, California becomes unavailable, the west-coast customers must be served from facilities in Springfield, Illinois and Austin, Texas (Figure 2), resulting in a transportation cost of $1,081,000, an increase of 112%. The "failure costs" (the transportation cost when a site fails) of the five optimal facilities, as well as their assigned demands, are listed in Table 1, as is the transportation cost when no facilities fail. Note that Sacramento serves a relatively small portion of the demand; its large failure cost is due to its distance from good "backup" facilities. In contrast, Harrisburg, Pennsylvania, is relatively close to two good backup facilities, but because it serves one-third of the total demand, its failure, too, is very costly. Springfield is the second-largest facility in terms of demand served, but its failure cost is much smaller because it is centrally located, close to good backup facilities. The reliability of a facility, then, can depend either on the distance from other facilities (e.g., Sacramento, which is small but distant) or on the demand served (Harrisburg, which is close but large), or on both

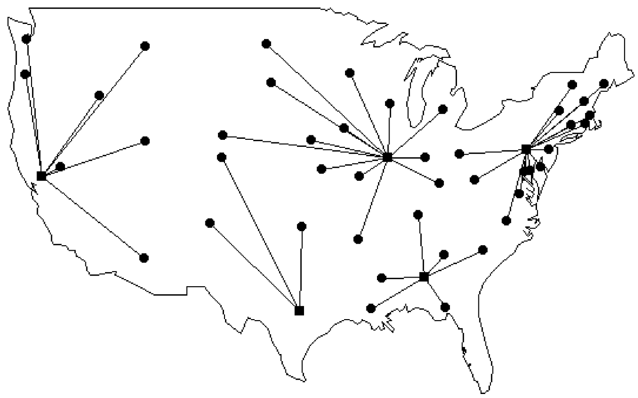

Figure 1 **UFLP Solution to 49-Node Dataset**

(Springfield, which is reliable because it is neither excessively large nor excessively distant).

A more reliable solution locates facilities in the capitals of California, New York, Texas, Pennsylvania, Ohio, Alabama, Oregon, and Iowa; in this solution, no facility has a failure cost of more than $640,000, rivaling the smallest failure cost in Table 1. On the other hand, three additional facilities are used in this solution, and these come at a cost. Few firms would be willing to choose solutions with location and day-to-day transportation costs that are much greater than optimal just to hedge against occasional and unpredictable disruptions in their supply network. One of the goals of this paper is to demonstrate that substantial improvements in reliability can often be obtained without large increases in day-to-day operating cost—that by taking reliability into account at design time, one can find a near-optimal UFLP solution that is much more reliable. This is demonstrated by examining the trade-off between the operating cost and the *expected* failure cost of the system, given a probability that each facility will fail and assuming that multiple facilities can fail simultaneously. One may instead wish to consider the *maximum* failure cost among all

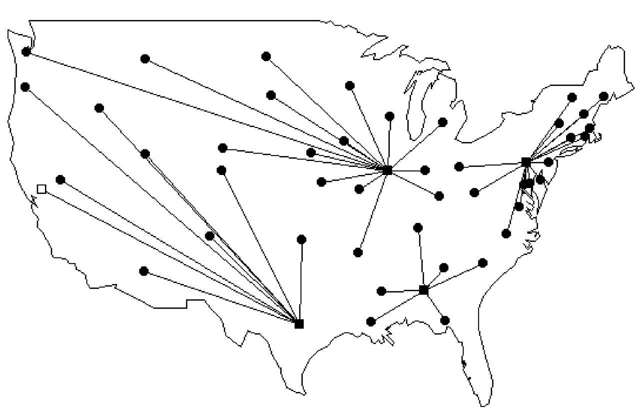

Figure 2 **UFLP Solution to 49-Node Dataset, After Failure of Facility in Sacramento**

Table 1 **Failure Costs and Assigned Demands for UFLP Solution**

Location	% Demand served	Failure cost	% Increase
Sacramento, CA	19	1,081,229	112
Harrisburg, PA	33	917,332	80
Springfield, IL	22	696,947	37
Montgomery, AL	16	639,631	26
Austin, TX	10	636,858	25
Transp. cost w/no failures		508,858	0

facilities, rather than the expected cost; this measure is treated by Snyder (2003).

This reliability approach is new in the logistics literature. It differs from traditional approaches to optimization under uncertainty, in which the goal is to choose a solution that performs well with respect to uncertain future conditions (e.g., random demands or costs). Our models seek solutions that perform well when parts of the system fail. In a sense, we are hedging against uncertainty in the solution itself. Another way of viewing these models is that, unlike stochastic facility location models that seek *demand-side* robustness (robustness to changes in demand or costs), these models seek *supply-side* robustness (robustness to changes in the supply network itself).

We present reliability-based formulations of both the UFLP and the P-median problem, another classical facility location problem in which the number of facilities to be located is fixed. The remainder of this paper is structured as follows. We review the related literature in §2. In §3, we formulate a reliability model based on the P-median problem. We solve this model using Lagrangian relaxation and show how to use it to generate a trade-off curve between operating cost and expected failure cost using the weighting method of multiobjective programming. In §4, we extend this model to solve a reliability version of the UFLP and discuss a modification that results in much better computational bounds with little loss of accuracy. We present computational results in §5 and a summary in §6.

2. Literature Review

There are three main bodies of literature that are similar—in spirit, if not in modeling approach—to the research presented in this paper. The first is the literature on network reliability, most often applied to telecommunications or power transmission networks. The second concerns expected or backup covering models, which are frequently used in locating emergency services facilities or vehicles. Finally, our models can be seen as an outgrowth of a small body of literature that discusses approaches for handling disruptions to supply chains. We discuss each of these three research areas next.

The concept of reliability is borrowed from network reliability theory (Colbourn 1987; Shier 1991; Shooman 2002), which is concerned with computing, estimating, or maximizing the probability that a network (typically a telecommunications or power network, represented by a graph) remains connected in the face of random failures. Failures may be due to disruptions, congestion, or blockages. Almost all of the research on network reliability considers failures only on the edges, but occasional papers consider node failures as well (e.g., Eiselt, Gendreau, and Laporte 1996). The network reliability literature tends to focus either on computing reliability or on optimizing it, i.e., either determining the reliability of a given system or designing a reliable system from scratch. Computing the reliability of a given network is a nontrivial problem (see, e.g., Ball 1979), and various performance measures and techniques for computing them have been proposed. Because of the complications involved in computing reliability, reliability optimization models rarely include explicit expressions for the reliability of the network. Instead, they often attempt to find the minimum-cost network design with some desired structural property, such as 2-connectivity (Monma and Shallcross 1989; Monma, Munson, and Pulleyblank 1990), k-connectivity (Bienstock, Brickell, and Monma 1990; Grötschel, Monma, and Stoer 1995), or special ring structures (Fortz and Labbé 2002). The key difference between network reliability models and the models that we present in this paper is that network models are concerned entirely with connectivity. The only costs considered are those to construct the network, not the transportation cost after rerouting, which is the primary concern of our reliability models.

Our models are also similar in spirit to the vector-assignment P-median problem (VAPMP) by Weaver and Church (1985) in that we assign customers to facilities at multiple levels. In the VAPMP, customers are assumed to be served by multiple facilities based on preference and availability. For example, a given customer might receive 80% of its demand from its nearest facility, 15% from its second-nearest, and 5% from its third-nearest. These percentages are inputs to the model. In our models, the "higher-level" assignments are only used when the primary facilities fail; there are no prespecified fractions of demand served by each facility. A similar model, based on the UFLP, is presented by Pirkul (1989).

Several papers extend the maximum covering problem (Church and ReVelle 1974) to handle the randomness inherent in locating emergency services vehicles. The classical maximum covering problem assumes that a vehicle is always available when a call for service arrives, but this fails to model the congestion in such systems when multiple calls are received by a facility with limited resources. Daskin (1982) formulates the maximum expected covering location model (MEXCLM), which assumes a constant, systemwide probability that a server is busy when a call is received and maximizes the total expected coverage; he solves the problem heuristically in Daskin (1983). ReVelle and Hogan (1989) present the maximum availability location problem (MALP), which allows the availability probability to vary among service areas, while Ball and Lin (1993) justify the form of the coverage constraints in MEXCLM and MALP using a system reliability approach.

Larson (1974, 1975) introduced queueing-based location models that explicitly consider customers waiting for service in congested systems. His "hypercube model" is useful as a descriptive model, but because of its complexity, researchers have had difficulty incorporating it into optimization models. Berman, Larson, and Chiu (1985) incorporate the hypercube idea into a simple optimization model, presenting theoretical results about the trajectory of the optimal 1-median as the demand rate changes in a general network.

Daskin, Hogan, and ReVelle (1988) compare various stochastic covering problems in which the objective is to locate facilities to maximize expected coverage or the degree of backup coverage. Berman and Krass (2001) consolidate a wide range of approaches to facility location in congested systems, presenting a complex model that is illustrative, but can be solved only for special cases. The key differences between the expected and backup coverage models and our models are (1) the objective function (coverage versus cost) and (2) the nature of the unavailability of a server (congestion versus failures).

Finally, we view our models as an outgrowth of the small body of literature, mainly appearing in response to the terrorist attacks on September 11, 2001, calling for techniques for designing and operating supply chains that are resilient to disruptions of all sorts. Articles appearing in academic journals (Sheffi 2001), business journals (Martha and Vratimos 2002; Simchi-Levi, Snyder, and Watson 2002; Navas 2003), and popular magazines (Lynn 2002) make compelling arguments that supply chains are particularly vulnerable to intentional or accidental disruptions and suggest possible approaches for making them less so, but they do not present any quantitative models. We view the present work as beginning to fill this void.

To our knowledge, the only analytical models considering failures in a facility location or supply chain design context are those of Krass, Berman, and Menezes (2003), Menezes, Berman, and Krass (2003a, b), and Bundschuh, Klabjan, and Thurston (2003). The first references assume that customers travel from facility to facility in search of an operational one (for example, when looking for an ATM); their models minimize a nonlinear objective representing the expected cost incurred by customers as they travel along these paths. Bundschuh, Klabjan, and Thurston (2003) choose suppliers in an inbound supply chain so that the resulting systems are "reliable" (have a low probability that any supplier fails) and/or "robust" (are able to maintain a high level of output even after suppliers have failed). Note that their definition of reliability is somewhat different from ours. Reliability is enforced by requiring the probability that all suppliers are operational to exceed a desired level; this is a multiechelon version of the model proposed by Vidal and Goetschalckx (2000). Systemwide reliability can be improved by switching to more reliable suppliers, but not by adding redundant suppliers: Increasing the number of suppliers *decreases* the reliability of the system because it increases the likelihood that one or more suppliers will fail. The robustness model allows nodes to keep emergency stock on hand and to obtain extra material from operational suppliers when one supplier has failed (though it ignores the additional cost that such procurement would entail). A joint reliability/robustness model combines the two approaches. The authors test their models on two moderately sized instances using a standard MIP solver with runtimes of up to one hour. They discuss both qualitative (e.g., shifting supply from Southeast Asia to North America) and quantitative (e.g., changes to the mean and standard deviation of output, measured using simulation) empirical differences between solutions from the different models.

3. The Reliability P-Median Problem

In this section we discuss a P-median-based model that minimizes a weighted sum of the *operating cost* (the day-to-day transportation cost when all facilities are operational) and the *expected failure cost* (the expected transportation cost, taking into account random facility failures). Each facility fails with a given probability, and multiple facilities may fail simultaneously. Certain facilities may be designated as "nonfailable." In our work with a major manufacturer of durable goods, the facilities that may fail represent warehouses owned by independent distributors who occasionally "defect" from the company or go out of business. The nonfailable warehouses are those owned by the company; these are assumed to remain loyal to the firm and will not fail. In other applications, the nonfailable facilities may represent those located in favorable weather areas, those served by unions with which the firm has a particularly strong relationship, or other facilities deemed to have a negligible probability of failure.

3.1. Formulation

Let I be the set of customers, indexed by i, and J the set of potential facility locations, indexed by j. Let NF be the set of candidate facilities that may not fail (we refer to these as "nonfailable" facilities) and let F be the set of candidate facilities that may fail ("failable" facilities). Note that $NF \cup F = J$. Each customer $i \in I$ has a demand h_i. The cost per unit of demand to ship from facility $j \in J$ to customer $i \in I$ is given by d_{ij}. Associated with each customer i is a cost θ_i that represents the cost of not serving the customer, per unit of demand. (θ_i may be a lost-sales cost, or the cost of serving i by purchasing product from a competitor on an emergency basis.) This cost is incurred if all open facilities have failed (and thus no facilities are available to serve customer i), or if θ_i is less than the cost of assigning i to any of the existing facilities that have not failed. To model this, we add an "emergency" facility u that is nonfailable ($u \in NF$) and has transportation cost $d_{iu} = \theta_i$ to customer $i \in I$. We force u to be open and replace P with $P+1$. From this point forward, we assume that the emergency facility has been handled in this way, though for simplicity we continue to formulate the problem as a P-median, rather than as a $(P+1)$-median problem.

Each facility in F has a probability q of failing, which is interpreted as the long-run fraction of time the facility is nonoperational. In some cases, q may be estimated based on historical data (e.g., for weather-induced failures), while in others q must be estimated subjectively (e.g., for failures due to defection of third-party distributors). Our model is most easily interpreted as an infinite-horizon model in which q represents the fraction of time that a facility has failed. However, if the modeler has in mind a particular time horizon T, then q may be used to capture probabilistic information about the timing of the failures. For example, suppose each facility has a 0.1 probability of failing, and that if it fails, it will fail during periods 1 through 5 with probability 0.3 and in periods 3 through T with probability 0.7. (Note that this means that if a facility fails, it will surely be inoperable during periods 3, 4, and 5.) Then the expected fraction of time the facility will be nonoperational is given by $(0.1 \times 0.3 \times 5 + 0.1 \times 0.7 \times (T-2))/T$.

The strategy behind our formulation is to assign each customer to a primary facility that will serve it under normal circumstances, as well as to a set of backup facilities that serve it when the primary facility has failed. Because multiple failures may occur simultaneously, each customer needs a first backup facility in case its primary facility fails, a second backup in case its first backup fails, and so on. However, if a customer is assigned to a nonfailable facility as its nth backup, it does not need any further backups.

There are two sets of decision variables in the model, location variables (X) and assignment variables (Y):

$$X_j = \begin{cases} 1, & \text{if a facility is opened at location } j \\ 0, & \text{otherwise.} \end{cases}$$

$$Y_{ijr} = \begin{cases} 1, & \text{if demand node } i \text{ is assigned to facility } j \text{ as a level-}r\text{ assignment} \\ 0, & \text{otherwise.} \end{cases}$$

A "level-r" assignment is one for which there are r closer failable facilities that are open. If $r = 0$, this is a primary assignment; otherwise, it is a backup assignment. Each customer i has a level-r assignment for each $r = 0, \ldots, P-1$, unless i is assigned to a level-s facility that is nonfailable, where $s < r$. In other words, customer i is assigned to one facility at level 0, another facility at level 1, and so on until i has been assigned to all open facilities at some level, or i has been assigned to a nonfailable facility.

We formulate this problem as a multiobjective problem. The objectives are as follows:

$$w_1 = \sum_{i\in I}\sum_{j\in J} h_i d_{ij} Y_{ij0}$$

$$w_2 = \sum_{i\in I} h_i \left[\sum_{j\in NF}\sum_{r=0}^{P-1} d_{ij} q^r Y_{ijr} + \sum_{j\in F}\sum_{r=0}^{P-1} d_{ij} q^r (1-q) Y_{ijr} \right].$$

Objective w_1 computes the operating cost—the P-median cost of serving customers from their primary facilities. Objective w_2 computes the expected failure cost: Each customer i is served by its level-r facility (call it j) if the r closer facilities have failed (this occurs with probability q^r) and if j itself has not failed (this occurs with probability $1-q$ if $j \in F$ and with probability 1 if $j \in NF$). Note that by the definition of level-r, all r closer facilities are failable.

Although we refer to w_2 as the "expected failure cost," we are careful to point out that w_2 also includes the transportation cost when no facilities have failed (i.e., the level-0 assignments). Certainly, there are ways to define reliability other than that given in w_2. For example, if the desired trade-off is between PMP cost and expected transportation cost only after a failure, then the "primary" transportation cost can be omitted from w_2 by starting the summation indices at $r = 1$ rather than $r = 0$. It is also possible that the transportation costs for backup assignments are different from those for primary assignments because, for example, they are arranged with freight companies on an emergency basis; in this case, the coefficients for Y_{ijr} would be changed from d_{ij} to some other cost for $r > 0$. Either of these modifications can be handled easily using the solution method described below.

Our model minimizes a weighted sum $\alpha w_1 + (1-\alpha) w_2$ of the two objectives, where $0 \le \alpha \le 1$. By solving the problem for various values of α, one can generate a trade-off curve between the operating cost and the expected failure cost using the weighting method of multiobjective programming (see §3.3). The decision maker can then choose a solution from the trade-off curve in accordance with his or her preference between the two objectives. Furthermore, the trade-off curve can indicate the degree to which one objective must be sacrificed to improve the other. In our empirical results, we show that the trade-off curve for the reliability P-median problem (RPMP) is "steep"—that large reductions in objective w_2 can be attained with only small increases in w_1.

The RPMP is formulated as follows:

(*RPMP*)

$$\text{minimize} \quad \alpha w_1 + (1-\alpha) w_2 \tag{1}$$

subject to

$$\sum_{j\in J} Y_{ijr} + \sum_{j\in NF}\sum_{s=0}^{r-1} Y_{ijs} = 1 \quad \forall\, i \in I,\ r = 0, \ldots, P-1 \tag{2}$$

$$Y_{ijr} \le X_j \quad \forall\, i \in I,\ j \in J,\ r = 0, \ldots, P-1 \tag{3}$$

$$\sum_{j\in J} X_j = P \tag{4}$$

$$\sum_{r=0}^{P-1} Y_{ijr} \le 1 \quad \forall\, i \in I,\ j \in J \tag{5}$$

$$X_u = 1 \tag{6}$$

$$X_j \in \{0, 1\} \quad \forall\, j \in J \tag{7}$$

$$Y_{ijr} \in \{0, 1\} \quad \forall\, i \in I,\ j \in J,\ r = 0, \ldots, P-1. \tag{8}$$

The objective function (1) is straightforward. Constraints (2) require that for each customer i and each level r, either i is assigned to a level-r facility or it is assigned to a level-s facility ($s < r$) that is nonfailable. (By convention we take $\sum_{s=0}^{r-1} Y_{ijs} = 0$ if $r = 0$.) Constraints (3) prohibit an assignment to a facility that has not been opened. Constraint (4) requires P facilities to be opened. Constraints (5) prohibit a customer from being assigned to a given facility at more than one level. Constraint (6) requires the emergency facility u to be opened. Constraints (7) and (8) are standard integrality constraints. (In fact, it is sufficient to require $Y_{ijr} \ge 0$ in place of (8). We use integrality constraints to emphasize the binary nature of the assignment decision; doing so does not affect our solution procedure.)

For notational convenience, we can write the objective function as

$$\sum_{i\in I}\sum_{j\in J}\sum_{r=0}^{P-1} \psi_{ijr} Y_{ijr}, \tag{9}$$

where

$$\psi_{ijr} = \begin{cases} \alpha h_i d_{ij} + (1-\alpha) h_i d_{ij} = h_i d_{ij}, & \text{if } r = 0 \text{ and } j \in NF \\ \alpha h_i d_{ij} + (1-\alpha) h_i d_{ij}(1-q), & \text{if } r = 0 \text{ and } j \in F \\ (1-\alpha) h_i d_{ij} q^r, & \text{if } r > 0 \text{ and } j \in NF \\ (1-\alpha) h_i d_{ij} q^r (1-q), & \text{if } r > 0 \text{ and } j \in F. \end{cases}$$

One might suspect that for small α, the weight on the backup assignments may be larger than that on the primary assignments, in which case it may be optimal to assign customers to primary facilities that are farther than their backup facilities, a situation we would want to prohibit. The next theorem, however, demonstrates that such a situation cannot occur.

Theorem 1. *In any optimal solution to* (*RPMP*), *if* $Y_{ijr} = Y_{ik,r+1} = 1$ *for* $i \in I$, $j, k \in J$, $0 \le r < P-1$, *then* $d_{ij} \le d_{ik}$.

Proof. Suppose, for a contradiction, that (X, Y) is an optimal solution to (*RPMP*) in which $Y_{ijr} = Y_{ik,r+1} = 1$, but $d_{ij} > d_{ik}$. We will show that by "swapping" j and k, the objective function will decrease. Because i has a level-$(r+1)$ facility (k), its level-r facility (j) must be failable.

Suppose first that $k \in F$. These two assignments contribute $\psi_{ijr} + \psi_{ik,r+1}$ to the objective function. If we assigned i to j at level $r+1$ and to k at level r, the objective function would change by $(\psi_{ikr} + \psi_{ij,r+1}) - (\psi_{ijr} + \psi_{ik,r+1})$. If $r = 0$, then

$$\begin{aligned}
&(\psi_{ikr} + \psi_{ij,r+1}) - (\psi_{ijr} + \psi_{ik,r+1}) \\
&\quad = \alpha h_i d_{ik} + (1-\alpha) h_i d_{ik}(1-q) + (1-\alpha) h_i d_{ij} q(1-q) \\
&\qquad - \alpha h_i d_{ij} - (1-\alpha) h_i d_{ij}(1-q) - (1-\alpha) h_i d_{ik} q(1-q) \\
&\quad = \alpha h_i (d_{ik} - d_{ij}) + (1-\alpha) h_i (d_{ik} - d_{ij})(1-q)^2 \\
&\quad < 0
\end{aligned}$$

because $d_{ij} > d_{ik}$ and $0 \le \alpha \le 1$. On the other hand, if $r > 0$, then

$$\begin{aligned}
&(\psi_{ikr} + \psi_{ij,r+1}) - (\psi_{ijr} + \psi_{ik,r+1}) \\
&\quad = (1-\alpha) h_i d_{ik} q^r (1-q) + (1-\alpha) h_i d_{ij} q^{r+1}(1-q) \\
&\qquad - (1-\alpha) h_i d_{ij} q^r (1-q) - (1-\alpha) h_i d_{ik} q^{r+1}(1-q) \\
&\quad = (1-\alpha) h_i (d_{ik} - d_{ij}) q^r (1-q)^2 \\
&\quad < 0.
\end{aligned}$$

Either way, the objective function is smaller for the revised solution. The case in which $k \in NF$ is similar, except that in this case, $Y_{ij,r+1} = 0$ because i's level-r facility is nonfailable, resulting in an even larger decrease in cost. This contradicts the assumption that (X, Y) is optimal. □

We note briefly that if the level-0 assignments are excluded from w_2 as discussed on page 404, then Theorem 1 holds when $\alpha \ge \frac{1}{2}$, which is generally the range of interest to decision makers. In this case, the algorithm given below may still be valid for particular instances, even if $\alpha < \frac{1}{2}$. If the algorithm returns a solution for which the distance ordering is obeyed, it is optimal; but the algorithm cannot enforce the distance ordering if it is not naturally optimal to do so.

3.2. Lagrangian Relaxation

3.2.1. Lower Bound. (*RPMP*) could be solved using an off-the-shelf IP solver, but in general such an approach will yield excessive runtimes, even for moderately sized problems. (See §5.3.) This motivates the development of a Lagrangian relaxation algorithm. Relaxing constraints (2) with Lagrange multipliers λ yields the following Lagrangian subproblem:

(*RPMP-LR*$_\lambda$)

$$\text{minimize } z(\lambda) = \sum_{i\in I}\sum_{j\in J}\sum_{r=0}^{P-1} \psi_{ijr} Y_{ijr} + \sum_{i\in I}\sum_{r=0}^{P-1} \lambda_{ir}\left(1 - \sum_{j\in J} Y_{ijr} - \sum_{j\in NF}\sum_{s=0}^{r-1} Y_{ijs}\right) \tag{10}$$

subject to

$$Y_{ijr} \le X_j \quad \forall\, i \in I,\ j \in J,\ r = 0, \ldots, P-1 \tag{11}$$

$$\sum_{j\in J} X_j = P \tag{12}$$

$$\sum_{r=0}^{P-1} Y_{ijr} \le 1 \quad \forall\, i \in I,\ j \in J \tag{13}$$

$$X_u = 1 \tag{14}$$

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

$$Y_{ijr} \in \{0, 1\} \quad \forall\, i \in I,\ j \in J,\ r = 0, \ldots, P-1. \tag{16}$$

The objective function (10) can be rewritten as follows:

$$\begin{aligned}
&\sum_{i\in I}\sum_{j\in J}\sum_{r=0}^{P-1} \psi_{ijr} Y_{ijr} + \sum_{i\in I}\sum_{r=0}^{P-1} \lambda_{ir} - \sum_{i\in I}\sum_{j\in J}\sum_{r=0}^{P-1} \lambda_{ir} Y_{ijr} \\
&\qquad - \sum_{i\in I}\sum_{r=0}^{P-1}\sum_{j\in NF}\sum_{s=0}^{r-1} \lambda_{ir} Y_{ijs} \\
&\quad = \sum_{i\in I}\sum_{j\in J}\sum_{r=0}^{P-1} \psi_{ijr} Y_{ijr} + \sum_{i\in I}\sum_{r=0}^{P-1} \lambda_{ir} - \sum_{i\in I}\sum_{j\in J}\sum_{r=0}^{P-1} \lambda_{ir} Y_{ijr} \\
&\qquad - \sum_{i\in I}\sum_{j\in NF}\sum_{s=0}^{P-1}\sum_{r=0}^{s-1} \lambda_{is} Y_{ijr}
\end{aligned}$$

(by swapping the indices r and s in the last term)

$$\begin{aligned}
&\quad = \sum_{i\in I}\sum_{j\in J}\sum_{r=0}^{P-1} \psi_{ijr} Y_{ijr} + \sum_{i\in I}\sum_{r=0}^{P-1} \lambda_{ir} - \sum_{i\in I}\sum_{j\in J}\sum_{r=0}^{P-1} \lambda_{ir} Y_{ijr} \\
&\qquad - \sum_{i\in I}\sum_{j\in NF}\sum_{\substack{r=0,\ldots,P-1\\ s=0,\ldots,P-1\\ r<s}} \lambda_{is} Y_{ijr}
\end{aligned}$$

$$= \sum_{i\in I}\sum_{j\in J}\sum_{r=0}^{P-1}\psi_{ijr}Y_{ijr} + \sum_{i\in I}\sum_{r=0}^{P-1}\lambda_{ir} - \sum_{i\in I}\sum_{j\in J}\sum_{r=0}^{P-1}\lambda_{ir}Y_{ijr}$$
$$- \sum_{i\in I}\sum_{j\in NF}\sum_{r=0}^{P-1}\left(\sum_{s=r+1}^{P-1}\lambda_{is}\right)Y_{ijr}.$$

Therefore, the objective function can be written as

$$\sum_{i\in I}\sum_{j\in J}\sum_{r=0}^{P-1}\tilde{\psi}_{ijr}Y_{ijr} + \sum_{i\in I}\sum_{r=0}^{P-1}\lambda_{ir}, \tag{17}$$

where

$$\tilde{\psi}_{ijr} = \begin{cases} \psi_{ijr} - \lambda_{ir}, & \text{if } j\in F \\ \psi_{ijr} - \lambda_{ir} - \left(\sum_{s=r+1}^{P-1}\lambda_{is}\right) = \psi_{ijr} - \sum_{s=r}^{P-1}\lambda_{is}, & \text{if } j \in NF. \end{cases} \tag{18}$$

For given λ, problem (*RPMP-LR*$_\lambda$) can be solved easily. Because the assignment constraints (2) have been relaxed, customer i may be assigned to zero, one, or more than one open facility at each level, but it may not be assigned to a given facility at more than one level r. Suppose that facility j is opened. Customer i will be assigned to facility j at level r if $\tilde{\psi}_{ijr} < 0$ and $\tilde{\psi}_{ijr} \le \tilde{\psi}_{ijs}$ for all $s = 0, \ldots, P-1$. Therefore, the *benefit* of opening facility j (i.e., the contribution to the objective function if j is opened) is given by

$$\gamma_j = \sum_{i\in I}\min\left\{0, \min_{r=0,\ldots,P-1}\{\tilde{\psi}_{ijr}\}\right\}. \tag{19}$$

Once the benefits γ_j have been computed for all j, we set $X_j = 1$ for the emergency facility u and for the $P-1$ remaining facilities with the smallest γ_j; we set $Y_{ijr} = 1$ if (1) facility j is open, (2) $\tilde{\psi}_{ijr} < 0$, and (3) r minimizes $\tilde{\psi}_{ijs}$ for $s = 0, \ldots, P-1$. The optimal objective value for (*RPMP-LR*$_\lambda$) is

$$z(\lambda) = \sum_{j\in J}\gamma_j X_j + \sum_{i\in I}\sum_{r=0}^{P-1}\lambda_{ir},$$

and this provides a lower bound on the optimal objective value of (*RPMP*).

The benefit γ_j can be computed for a single j in $O(nP)$ time, where $n = |I|$, so all of the benefits can be computed in $O(mnP)$ time, where $m = |J|$. Determining X_j requires sorting the facilities, which takes $O(m\log m)$ time, and determining Y_{ijr} requires $O(nP)$ time, assuming that assignments are stored as a single index j for each i, r rather than as a list of m 0/1 variables. Therefore, the Lagrangian subproblem can be solved for a given λ in $O(mnP + m\log m + nP) = O(mnP)$ time.

3.2.2. Upper Bound. At each iteration of the Lagrangian process, we obtain both a lower and an upper bound. The solution to (*RPMP-LR*$_\lambda$) provides a lower bound. If it is feasible for (*RPMP*), then it provides an upper bound as well, and is in fact optimal for (*RPMP*); because the constraint violations in (10) equal 0, the lower and upper bounds are equal. If the solution to (*RPMP-LR*$_\lambda$) is not feasible for (*RPMP*), as is the case in most iterations, then we construct a feasible solution by opening the facilities that are open in the solution to (*RPMP-LR*$_\lambda$) and assigning customers to the open facilities level by level in increasing order of distance, until a nonfailable facility is assigned. (By Theorem 1, this is an optimal strategy for assigning customers to a given set of facilities, though the facilities themselves may not be optimal.) Anecdotally, we can report that the heuristic as described here has performed well in our computational tests, finding the optimal solution very quickly (generally within the first 100 Lagrangian iterations), though we have not explicitly recorded the iteration at which the optimal solution is found.

3.2.3. Multiplier Updating. Each value of λ provides a lower bound $z(\lambda)$ on the optimal objective value of (*RPMP*). To find the best possible lower bound, we need to solve

$$\underset{\lambda\in\mathbb{R}^{nP}}{\text{maximize}}\ z(\lambda). \tag{20}$$

This problem is solved approximately using subgradient optimization, applied in a straightforward manner as described by Fisher (1981, 1985) and Daskin (1995). In particular, at each iteration n we compute a step-size t^n as

$$t^n = \frac{\beta^n(\text{UB} - \mathcal{L}^n)}{\sum_{i\in I}\sum_{r=0}^{P-1}\left(1 - \sum_{j\in J}Y_{ijr} - \sum_{j\in NF}\sum_{s=0}^{r-1}Y_{ijs}\right)^2}, \tag{21}$$

where β^n is a constant at iteration n, initialized to 2 and halved when 30 consecutive iterations fail to improve the lower bound; $\mathcal{L}^n$ is the value of $z(\lambda)$ found at iteration n; and UB is the best known upper bound. The multipliers are updated by setting

$$\lambda_{ir}^{n+1} \leftarrow \lambda_{ir}^n + t^n\left(1 - \sum_{j\in J}Y_{ijr} - \sum_{j\in NF}\sum_{s=0}^{r-1}Y_{ijs}\right). \tag{22}$$

The Lagrangian process terminates when any of the following criteria are met:

- $(\text{UB} - \mathcal{L}^n)/\mathcal{L}^n < \epsilon$, for some optimality tolerance ϵ specified by the user;
- $n > n_{\max}$, for some iteration limit $n_{\max}$;
- $\beta^n < \beta_{\min}$, for some β limit $\beta_{\min}$.

3.2.4. Initial Multipliers. Our algorithm, like many Lagrangian relaxation algorithms, is somewhat sensitive to the choice of the initial Lagrange multipliers. To develop a strategy for computing good starting multipliers, we first examined the final multipliers for problems that had been solved to optimality. We discovered that the final multipliers for $r=0$ were roughly on the order of magnitude of 0.01 times the demand-weighted distance from each customer to its assigned facility, and that the optimal multipliers decreased roughly by an order of magnitude as r increased. Therefore, we settled on the following formula for the initial multipliers:

$$\lambda_{ir} = h_i \bar{d}/10^{r+2},$$

where $\bar{d} = \sum_{i\in I}\sum_{j\in J} d_{ij}/|I||J|$ is the average distance between pairs of customers and facilities.

3.2.5. Branch and Bound. If the Lagrangian process terminates with the lower and upper bounds equal (to within ϵ), an ϵ-optimal solution has been found and the algorithm terminates. Otherwise, we use branch and bound to close the optimality gap. We branch on the X_j (location) variables. At each branch-and-bound node, the facility selected for branching is the unfixed open facility with the greatest assigned demand. X_j 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 within ϵ of the objective function value of the best feasible solution found anywhere in the tree, if P facilities have been forced open, or if $|J|-P$ facilities have been forced closed. The final Lagrange multipliers at a given node are passed to its child nodes and are used as initial multipliers at those nodes.

3.2.6. 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. Assume for notational convenience that the facilities in $J\backslash\{u\}$ are sorted in increasing order of benefit so that $\gamma_j \le \gamma_{j+1}$, under a particular set of Lagrange multipliers λ. Let LB be the lower bound (the objective value of (*RPMP-LR*$_\lambda$)) under the same λ, and let UB be the best upper bound found. Suppose further that $X_j = 0$ in the solution to (*RPMP-LR*$_\lambda$). If

$$\text{LB} + \gamma_j - \gamma_{P-1} > \text{UB}, \tag{23}$$

then candidate site j *cannot* be part of the optimal solution, so we can fix $X_j=0$. This is true because if j were forced into the solution, another facility would be forced out; this facility would be the open facility (other than u) with the largest benefit, i.e., facility $P-1$. Clearly, $\text{LB}+\gamma_j-\gamma_{P-1}$ 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 (*RPMP-LR*$_\lambda$). If

$$\text{LB} - \gamma_j + \gamma_P > \text{UB}, \tag{24}$$

then candidate site j *must* be part of the optimal solution because swapping j out, and the best closed facility in, will result in a solution whose lower bound exceeds the upper bound; therefore, we can fix $X_j=1$.

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 the Lagrangian procedure tends to produce tight lower bounds, making (23) or (24) hold for many facilities j. The time required to perform these checks is negligible. Variable fixing tends not to result in substantial improvements if performed at lower levels of the branch-and-bound tree, so we perform the procedure only at the root node.

3.3. Trade-off Curve

By systematically varying the objective function weight α and re-solving (*RPMP*) for each value, one can generate a trade-off curve between the two objectives using the weighting method of multiobjective programming (Cohon 1978). The method is as follows.

(0) Solve (*RPMP*) for $\alpha=1$ (the pure PMP problem) and for $\alpha=0$. Add both points to the trade-off curve.

(1) Identify a pair of adjacent solutions on the trade-off curve that has not yet been considered. Let the objective values of these two solutions be (w_1^1, w_2^1) and (w_1^2, w_2^2). Set $\alpha \leftarrow -(w_2^1-w_2^2)/(w_1^1-w_1^2-w_2^1+w_2^2)$.

(2) Solve (*RPMP*) for the current value of α. If the resulting solution is not already on the trade-off curve, add it.

(3) If all pairs of adjacent solutions on the trade-off curve have been explored, stop. Else, go to 1.

4. The Reliability Fixed-Charge Location Problem

The RPMP can improve reliability only by choosing a different set of P facilities, not by opening additional ones. In this section, we formulate the reliability fixed-charge location problem (RFLP), which is based on the UFLP. As the UFLP does not contain a limit on the number of facilities that can be built, the RFLP adds an additional degree of freedom for improving reliability, namely, constructing additional facilities.

4.1. Formulation

The RFLP is formulated in a manner similar to the RPMP. We need one additional parameter: f_j is the fixed cost to construct a facility at location $j \in J$, amortized to the time units used to express demands. Because the number of facilities is not known a priori as it is in the RPMP, we must create assignment variables for levels $r = 0, \dots, m-1$, where $m \equiv |J|$. The objectives are given by

$$w_1 = \sum_{j\in J} f_j X_j + \sum_{i\in I}\sum_{j\in J} h_i d_{ij} Y_{ij0}$$

$$w_2 = \sum_{i\in I} h_i \left[\sum_{j\in NF}\sum_{r=0}^{m-1} d_{ij} q^r Y_{ijr} + \sum_{j\in F}\sum_{r=0}^{m-1} d_{ij} q^r (1-q) Y_{ijr} \right].$$

The emergency facility u is handled as in the RPMP, described in §3.1; it has no fixed cost ($f_u = 0$).

The RFLP is formulated as follows.

(*RFLP*)

$$\text{minimize } \alpha w_1 + (1-\alpha) w_2 \tag{25}$$

subject to

$$\sum_{j\in J} Y_{ijr} + \sum_{j\in NF}\sum_{s=0}^{r-1} Y_{ijs} = 1 \quad \forall i \in I,\ r = 0, \dots, m-1 \tag{26}$$

$$Y_{ijr} \le X_j \quad \forall i \in I,\ j \in J,\ r = 0, \dots, m-1 \tag{27}$$

$$\sum_{r=0}^{m-1} Y_{ijr} \le 1 \quad \forall i \in I,\ j \in J \tag{28}$$

$$X_u = 1 \tag{29}$$

$$X_j \in \{0,1\} \quad \forall j \in J \tag{30}$$

$$Y_{ijr} \in \{0,1\} \quad \forall i \in I,\ j \in J,\ r = 0, \dots, m-1. \tag{31}$$

The formulation is identical to that of (*RPMP*) except:

- Fixed costs are included in objective w_1;
- Constraint (4) is omitted;
- The "level" index r is extended to $m-1$ instead of $P-1$ in summations and constraint indices.

Constraint (29) is not strictly necessary because facility u has 0 fixed cost, but including the constraint in the formulation tightens the Lagrangian relaxation. As in (*RPMP*), constraints (31) could be relaxed to $Y_{ijr} \ge 0$. Note that Theorem 1 applies to the RFLP as well.

4.2. Solution Method

To solve (*RFLP*), we relax constraints (26) to obtain the following Lagrangian subproblem:

(*RFLP-LR*$_\lambda$)

$$\text{minimize } z(\lambda) = \alpha \sum_{j\in J} f_j X_j + \sum_{i\in I}\sum_{j\in J}\sum_{r=0}^{m-1} \tilde{\psi}_{ijr} Y_{ijr} + \sum_{i\in I}\sum_{r=0}^{m-1} \lambda_{ir} \tag{32}$$

subject to

$$Y_{ijr} \le X_j \quad \forall i \in I,\ j \in J,\ r = 0, \dots, m-1 \tag{33}$$

$$\sum_{r=0}^{m-1} Y_{ijr} \le 1 \quad \forall i \in I,\ j \in J \tag{34}$$

$$X_u = 1 \tag{35}$$

$$X_j \in \{0,1\} \quad \forall j \in J \tag{36}$$

$$Y_{ijr} \in \{0,1\} \quad \forall i \in I,\ j \in J,\ r = 0, \dots, m-1. \tag{37}$$

In the objective function (32),

$$\tilde{\psi}_{ijr} = \begin{cases} \psi_{ijr} - \lambda_{ir}, & \text{if } j \in F \\ \psi_{ijr} - \lambda_{ir} - \left(\sum_{s=r+1}^{m-1} \lambda_{is} \right) = \psi_{ijr} - \sum_{s=r}^{m-1} \lambda_{is}, & \text{if } j \in NF. \end{cases} \tag{38}$$

The benefit γ_j of opening facility j is computed as

$$\gamma_j = \alpha f_j + \sum_{i\in I} \min\left\{0, \min_{r=0,\dots,m-1} \{\tilde{\psi}_{ijr}\}\right\}. \tag{39}$$

X_u is set to 1, and for $j \ne u$, X_j is set to 1 if $\gamma_j < 0$ (or if $\gamma_k \ge 0$ for all $k \in J$ but is smallest for j, because at least one "real" facility (excluding u) must be open in any feasible solution to the problem); Y_{ijr} is set following the criteria described in §3.2.1.

At each Lagrangian iteration, we find an upper bound by opening the facilities that are open in the solution to (*RFLP-LR*$_\lambda$) and greedily assigning customers to them. In addition, we perform an "add" and a "drop" heuristic on each solution whose objective value is less than $1.2 \times \text{UB}$, where UB is the best known upper bound. The add (drop) heuristic considers opening (closing) facilities if doing so decreases the objective value. Each heuristic is performed until no further adds or drops will improve the solution.

The subgradient optimization and branch-and-bound procedures are exactly as described for the RPMP, except that branch-and-bound nodes are fathomed if the lower bound at that node is within ϵ of the best known upper bound, if $|J|$ (rather than P) facilities have been forced open, or if $|J| - 1$ (rather than $|J| - P$) facilities have been forced closed.

4.3. A Modification

In our preliminary computational testing, we found that the subgradient optimization procedure had difficultly converging to a tight lower bound for the RFLP. We believe the problem results from the large number of multipliers that must be updated (nm of them, as opposed to nP in the RPMP). To counteract this effect, we propose the following modification of our model and algorithm. Because the probability of many facilities failing simultaneously is small, ignoring the simultaneous failure of more than, say, five facilities may result in a very small loss of accuracy. At the same time, the reduction in the number of multipliers may result in a very large improvement in computational performance. Customers would only be assigned to facilities at levels 0 through 4, and higher-level assignments would not be included either in the objective function or in the constraints. In fact, if we interpret m as the number of levels to be assigned, rather than as the cardinality of J, then the objective functions w_1 and w_2 and the formulation of (*RFLP*) remain intact under this new modeling scheme, as does the Lagrangian relaxation ($RFLP\text{-}LR_\lambda$) and the algorithm for solving it. The emergency facility may become irrelevant in this case, as it is generally used only when all open facilities have failed, but it may still play a role in the solution if the emergency cost is smaller than the cost of serving a given customer from, say, its fourth-nearest facility when the first three have failed.

We observed similar convergence problems in the RPMP when P was large. The same modification may be made to (*RPMP*) by replacing P with m (except in constraint (4)). We have found this modification to be very effective for both problems; our computational experience with this modification is presented in §5.5.

5. Computational Results

5.1. Experimental Design

We tested our algorithms on five datasets with up to 150 nodes.[1] The first three (see Daskin 1995) are derived from 1990 census data: a 49-node set consisting of the state capitals of the continental United States plus Washington, D.C.; an 88-node set consisting of the 49-node set plus the 50 largest cities in the United States, minus duplicates; and a 150-node set consisting of the 150 largest cities in the United States. Demands h_i are set to the state population divided by 10^5 for the 49-node set and to the city population divided by 10^4 for the other two. The fixed cost f_j is set to the median home value in the city for the 49- and 88-node sets and to 10^5 for all j in the 150-node set. The transportation cost d_{ij} is set equal to the great-circle distance between i and j. The emergency cost θ_i is set to 10^4 for all i.

The remaining two datasets were generated randomly. One dataset consists of 50 nodes; the other consists of 100 nodes. In both cases, demands h_i were drawn from $U[0, 1000]$ and rounded to the nearest integer, fixed costs f_j were drawn from $U[500, 1500]$ and rounded to the nearest integer, and x and y coordinates were drawn from $U[0, 1]$. Transportation costs d_{ij} are set equal to the Euclidean distance between i and j. The emergency cost θ_i is set to 10

[1] All datasets may be obtained from the lead author's website.

Table 2 Parameters for Lagrangian Relaxation Procedure

Parameter	Value
Optimality tolerance (ϵ)	0.001
Maximum number of iterations ($n_{\max}$) at root node	1,200
Maximum number of iterations ($n_{\max}$) at child nodes	600
Initial value of β	2
Number of nonimproving iterations before halving β	30
Minimum value of β ($\beta_{\min}$)	10^{-8}
Initial value for λ_{ir}	$\mu_i \bar{d}/10^{r+2}$

to model the situation in which losing a customer is extremely costly.

In all five datasets, the set I of customers and the set J of facilities are equal (i.e., every customer may become a distribution center (DC)). q was set to 0.05. We tested the RPMP algorithm for $P = 5, 10$, and 20, as well as the RFLP algorithm, using six different values of α. (We did not use the weighting method described in §3.3 in these initial tests; see §5.4 for results using the weighting method.) We executed the Lagrangian relaxation/branch-and-bound process to an optimality tolerance of 0.1%, or until 600 seconds (10 minutes) of CPU time had elapsed. The algorithm was coded in C++ and tested on a Gateway Profile 4MX desktop computer with a Pentium IV 3.2 GHz processor and 1.0 GB RAM, running under Windows XP. Parameter values for the Lagrangian relaxation algorithm are given in Table 2. (The notation $\bar{d}$ in the last line stands for $\sum_{i\in I}\sum_{j\in J} d_{ij}/|I||J|$.) The number of levels included in the objective function and constraints (m; see §4.3) was set to 5; the choice of m is explored further in §5.5.

5.2. Algorithm Performance

Table 3 summarizes the results for the RPMP, Table 4 for the RFLP. The Overall LB, UB, and Gap columns give the lower and upper bounds and the percentage gap, while the columns marked Root LB, UB, and Gap give the lower and upper bounds and the gap at the root node. The column marked # Lag iter. gives the total number of Lagrangian iterations, # BB nodes gives the total number of branch-and-bound nodes, and CPU time gives the total number of CPU seconds required. In Table 4, the column marked # Open indicates the number of facilities opened in the optimal solution (excluding the "emergency" facility u).

The algorithm solved 112 out of the 120 problems tested (93%) to optimality within the 600-second limit. The algorithm produced sufficiently tight bounds at the root node and required no branching to prove optimality for 51 (43%) of the problems. For 99 of the problems (83%), the optimal solution was found at the root node, even if optimality could not be proved without branching for some of these. Of the 112 problems solved to optimality, the average percentage difference between the root-node UB and the final UB

Table 3 Algorithm Results: RPMP

Nodes	P	α	Overall LB	Overall UB	Overall gap (%)	Root LB	Root UB	Root gap (%)	# Lag iter.	# BB nodes	CPU time
49	5	1.0	502,233	502,732	0.10	502,233	502,732	0.10	893	1	3.6
49	5	0.8	517,694	518,210	0.10	517,694	518,210	0.10	149	1	0.7
49	5	0.6	533,158	533,687	0.10	533,158	533,687	0.10	175	1	0.8
49	5	0.4	547,760	548,279	0.09	547,760	548,279	0.09	308	1	1.3
49	5	0.2	561,877	562,437	0.10	561,877	562,437	0.10	139	1	0.6
49	5	0.0	575,577	576,153	0.10	575,577	576,153	0.10	87	1	0.4
49	10	1.0	275,430	275,701	0.10	274,143	275,701	0.57	2,557	7	10.8
49	10	0.8	283,320	283,601	0.10	283,320	283,601	0.10	592	1	2.5
49	10	0.6	291,215	291,501	0.10	291,215	291,501	0.10	471	1	2.0
49	10	0.4	299,109	299,402	0.10	299,109	299,402	0.10	776	1	3.2
49	10	0.2	306,996	307,302	0.10	306,996	307,302	0.10	416	1	1.7
49	10	0.0	314,889	315,202	0.10	314,889	315,202	0.10	334	1	1.4
49	20	1.0	113,225	113,330	0.09	82,862	113,852	37.40	31,018	87	139.6
49	20	0.8	119,543	119,663	0.10	106,923	122,780	14.83	11,183	27	51.6
49	20	0.6	125,870	125,995	0.10	120,449	128,150	6.39	15,099	37	70.2
49	20	0.4	132,203	132,328	0.09	128,782	133,784	3.88	5,454	15	24.8
49	20	0.2	138,523	138,661	0.10	127,997	138,704	8.37	8,170	19	37.2
49	20	0.0	144,783	144,926	0.10	140,842	145,767	3.50	8,450	21	39.3
88	5	1.0	873,996	874,859	0.10	873,996	874,859	0.10	436	1	4.7
88	5	0.8	900,827	901,706	0.10	900,827	901,706	0.10	389	1	4.2
88	5	0.6	927,662	928,554	0.10	927,662	928,554	0.10	103	1	1.3
88	5	0.4	954,455	955,402	0.10	954,455	955,402	0.10	198	1	2.2
88	5	0.2	981,308	982,249	0.10	981,308	982,249	0.10	269	1	3.0
88	5	0.0	1,003,250	1,004,250	0.10	1,003,250	1,004,250	0.10	122	1	1.5
88	10	1.0	511,663	512,174	0.10	508,454	512,174	0.73	4,232	11	45.1
88	10	0.8	525,170	525,694	0.10	524,923	525,694	0.15	1,349	3	13.0
88	10	0.6	538,678	539,215	0.10	538,376	539,215	0.16	1,279	3	12.7
88	10	0.4	552,186	552,735	0.10	552,186	552,735	0.10	286	1	2.9
88	10	0.2	565,700	566,256	0.10	565,700	566,256	0.10	839	1	8.0
88	10	0.0	579,190	579,761	0.10	579,137	579,761	0.11	1,369	3	13.5
88	20	1.0	233,427	250,125	7.15	176,156	263,303	49.47	55,552	132	>600.0
88	20	0.8	259,784	260,039	0.10	254,057	271,712	6.95	38,714	93	440.4
88	20	0.6	269,685	269,953	0.10	254,396	275,915	8.46	7,959	17	92.3
88	20	0.4	279,589	279,867	0.10	273,182	282,220	3.31	28,058	61	340.5
88	20	0.2	289,048	289,330	0.10	279,108	290,975	4.25	37,680	91	436.6
88	20	0.0	298,422	298,720	0.10	289,268	299,303	3.47	13,700	29	159.5
150	5	1.0	1,196,980	1,198,160	0.10	1,196,980	1,198,160	0.10	260	1	8.0
150	5	0.8	1,213,000	1,214,190	0.10	1,213,000	1,214,190	0.10	364	1	10.8
150	5	0.6	1,225,190	1,226,190	0.08	1,225,190	1,226,190	0.08	190	1	6.1
150	5	0.4	1,231,040	1,232,270	0.10	1,231,040	1,232,270	0.10	325	1	9.7
150	5	0.2	1,233,870	1,234,390	0.04	1,233,870	1,234,390	0.04	312	1	9.4
150	5	0.0	1,233,910	1,234,390	0.04	1,233,910	1,234,390	0.04	276	1	8.5
150	10	1.0	738,461	739,200	0.10	659,817	739,200	12.03	8,919	19	250.2
150	10	0.8	754,318	755,072	0.10	749,869	755,072	0.69	8,663	19	243.9
150	10	0.6	766,172	766,938	0.10	762,862	769,430	0.86	13,438	27	394.1
150	10	0.4	771,397	772,152	0.10	769,677	772,152	0.32	2,641	5	77.5
150	10	0.2	773,701	774,243	0.07	772,587	774,243	0.21	2,904	5	83.7
150	10	0.0	773,477	774,243	0.10	773,477	774,243	0.10	556	1	14.7
150	20	1.0	137,992	380,309	175.60	137,992	380,309	175.60	21,282	50	>600.0
150	20	0.8	349,152	382,366	9.51	349,152	386,197	10.61	19,074	40	>600.0
150	20	0.6	383,377	383,749	0.10	375,513	387,837	3.28	9,586	21	295.5
150	20	0.4	387,808	388,180	0.10	387,791	388,180	0.10	4,301	9	132.9
150	20	0.2	391,763	392,127	0.09	347,132	393,858	13.46	7,416	17	208.0
150	20	0.0	394,907	395,299	0.10	344,463	397,887	15.51	9,045	19	253.4

Table 3 (cont'd.)

Nodes	P	α	Overall LB	Overall UB	Overall gap (%)	Root LB	Root UB	Root gap (%)	# Lag iter.	# BB nodes	CPU time
50	5	1.0	3,209	3,212	0.10	3,209	3,212	0.10	982	1	4.2
50	5	0.8	3,261	3,264	0.10	3,261	3,264	0.10	501	1	2.2
50	5	0.6	3,312	3,316	0.10	3,312	3,316	0.10	640	1	2.8
50	5	0.4	3,364	3,367	0.10	3,364	3,367	0.10	494	1	2.2
50	5	0.2	3,409	3,413	0.10	3,409	3,413	0.10	416	1	1.8
50	5	0.0	3,455	3,458	0.09	3,455	3,458	0.09	279	1	1.2
50	10	1.0	1,644	1,645	0.09	723	1,645	127.64	9,523	23	44.0
50	10	0.8	1,687	1,689	0.10	1,338	1,689	26.22	6,007	13	27.3
50	10	0.6	1,730	1,732	0.10	1,710	1,732	1.28	2,361	5	10.8
50	10	0.4	1,774	1,776	0.09	1,774	1,776	0.09	1,043	1	4.5
50	10	0.2	1,817	1,819	0.10	1,783	1,819	2.04	2,533	5	11.4
50	10	0.0	1,861	1,863	0.10	1,847	1,863	0.85	1,675	3	7.4
50	20	1.0	719	720	0.10	−4,852	734	—	36,595	97	173.0
50	20	0.8	745	746	0.10	−1,696	746	—	12,550	29	58.8
50	20	0.6	772	773	0.10	−5,728	789	—	19,179	45	92.1
50	20	0.4	799	800	0.10	−2,124	800	—	17,184	41	88.2
50	20	0.2	825	826	0.10	−54	826	—	14,738	33	71.5
50	20	0.0	852	853	0.10	−2,757	853	—	18,610	45	88.9
100	5	1.0	8,351	8,359	0.09	8,324	8,359	0.41	3,266	7	41.1
100	5	0.8	8,466	8,474	0.10	8,446	8,474	0.33	2,557	5	31.5
100	5	0.6	8,577	8,586	0.10	8,565	8,586	0.24	1,414	3	17.3
100	5	0.4	8,688	8,697	0.10	8,683	8,697	0.16	1,364	3	16.9
100	5	0.2	8,800	8,808	0.10	8,800	8,808	0.10	346	1	4.5
100	5	0.0	8,911	8,920	0.10	8,911	8,920	0.10	193	1	2.5
100	10	1.0	4,858	4,863	0.10	4,218	4,863	15.28	6,775	17	84.7
100	10	0.8	4,947	4,952	0.10	4,910	4,952	0.85	2,124	5	27.0
100	10	0.6	5,036	5,041	0.10	5,025	5,041	0.31	1,651	3	20.3
100	10	0.4	5,125	5,130	0.09	5,125	5,130	0.09	570	1	6.9
100	10	0.2	5,210	5,215	0.10	5,143	5,215	1.39	3,421	9	45.0
100	10	0.0	5,289	5,294	0.10	5,149	5,294	2.80	3,456	7	45.9
100	20	1.0	27	2,776	10,041.40	27	2,776	10,041.40	45,607	102	> 600.0
100	20	0.8	2,816	2,835	0.69	−384	2,835	—	42,301	97	> 600.0
100	20	0.6	461	2,895	528.56	461	2,896	528.84	39,946	96	> 600.0
100	20	0.4	2,947	2,950	0.10	1,012	2,955	191.88	38,700	91	574.9
100	20	0.2	3,003	3,005	0.07	718	3,005	318.51	31,762	71	413.2
100	20	0.0	3,056	3,059	0.10	828	3,059	269.48	34,386	81	475.5

is 0.19%, suggesting that even without branching, the Lagrangian procedure can be relied upon to produce good feasible solutions.

The root-node lower bound was quite weak (sometimes even negative) for some of the problems. We believe this is due primarily to the choice of the initial Lagrange multipliers, and that with additional experimentation, one could find better starting multipliers for a given problem. Indeed, the branch-and-bound process generally improved the multipliers quickly, and all but a few of these problems were solved to optimality within the 600-second limit. For the few problems for which the branch-and-bound process executed for the full 600 seconds without improving the bounds, the "right-hand" branch of the tree was never explored, preventing the overall lower bound from being updated; nevertheless, improvement *was* made in the lower bound in the left-hand branch.

Two trends are evident from Tables 3 and 4. First, the algorithm performs worse in terms of both gap sizes and CPU time when P is large. This is not surprising, because these problems have many more feasible solutions, and finding good Lagrange multipliers can be particularly difficult. Second, the algorithm often produces weak bounds when $\alpha = 1$. This is surprising at first because $\alpha = 1$ is the "easy" case—the pure PMP (or UFLP) problem. However, these pure problems have many extraneous variables and constraints; the extra variables have no bearing on the objective function, leading to a large number of optimal solutions that are difficult to prove optimal. Location problems with highly regular cost structures (e.g., many customers are equidistant from many facilities) are well known to be difficult to solve. Moreover, the poor performance when $\alpha = 1$ is not a problem because good algorithms already exist for the classical PMP and UFLP.

Table 4 Algorithm Results: RFLP

Nodes	α	Overall LB	Overall UB	Overall gap (%)	Root LB	Root UB	Root gap (%)	# Lag iter.	# BB nodes	CPU time	# Open
49	1.0	855,959	856,810	0.10	830,504	856,810	3.17	4,149	11	19.2	6
49	0.8	790,275	791,014	0.09	790,275	791,014	0.09	201	1	1.1	6
49	0.6	707,332	707,982	0.09	707,332	707,982	0.09	361	1	1.8	8
49	0.4	589,099	589,677	0.10	589,099	589,677	0.10	524	1	3.0	10
49	0.2	404,512	404,903	0.10	403,946	404,903	0.24	1,518	3	6.8	16
49	0.0	19,285	19,303	0.09	18,510	19,303	4.28	1,956	5	8.1	49
88	1.0	1,200,700	1,201,880	0.10	1,200,700	1,201,880	0.10	394	1	5.1	9
88	0.8	1,112,990	1,114,070	0.10	1,112,990	1,114,070	0.10	598	1	7.8	9
88	0.6	1,011,980	1,012,970	0.10	1,011,980	1,012,970	0.10	736	1	9.7	10
88	0.4	871,495	872,364	0.10	855,355	872,364	1.99	13,185	31	273.6	14
88	0.2	605,380	605,983	0.10	603,911	605,983	0.34	2,959	7	46.0	24
88	0.0	17,695	17,712	0.10	17,278	17,712	2.52	2,639	7	27.7	88
150	1.0	1,668,620	1,670,290	0.10	1,668,620	1,670,290	0.10	835	1	26.3	7
150	0.8	1,546,990	1,548,350	0.09	1,546,990	1,548,350	0.09	464	1	19.0	8
150	0.6	1,350,270	1,351,600	0.10	1,350,270	1,351,600	0.10	865	1	41.8	12
150	0.4	1,094,320	1,095,400	0.10	1,094,320	1,095,400	0.10	302	1	13.7	14
150	0.2	791,337	792,127	0.10	791,337	792,127	0.10	546	1	28.6	20
150	0.0	11,116	11,126	0.09	8,615	11,126	29.15	1,498	3	43.2	150
50	1.0	6,727	6,733	0.10	6,727	6,733	0.10	818	1	3.6	4
50	0.8	6,208	6,214	0.10	6,208	6,214	0.10	653	1	2.9	5
50	0.6	5,612	5,617	0.10	5,612	5,617	0.10	927	1	4.2	5
50	0.4	4,862	4,866	0.09	4,844	4,866	0.47	1,288	3	6.5	6
50	0.2	3,558	3,561	0.10	1,799	3,561	97.94	10,876	27	60.3	11
50	0.0	81	81	0.10	−10,676	81	—	15,833	43	70.6	50
100	1.0	11,482	11,493	0.10	11,067	11,493	3.85	10,965	27	169.7	7
100	0.8	10,623	10,633	0.09	10,623	10,633	0.09	522	1	8.6	8
100	0.6	9,574	9,584	0.10	9,387	9,584	2.09	3,945	9	62.0	9
100	0.4	5,025	8,333	65.83	5,025	8,333	65.83	22,219	50	> 600.0	11
100	0.2	3,406	6,231	82.94	3,406	6,231	82.94	8,503	17	> 600.0	17
100	0.0	132	133	0.10	−20,941	133	—	25,671	71	365.7	100

Finally, we note that, not surprisingly, the number of facilities opened in optimal solutions to the RFLP increases as α decreases, as for small α the emphasis is on reliability more than operational cost. In the extreme case in which $\alpha = 0$, the fixed costs are not included in the objective function at all, and the resulting solution is trivial: All facilities are open.

5.3. CPLEX Performance

To demonstrate the difficulty in solving (*RPMP*) and (*RFLP*) using an off-the-shelf IP solver, we solved an instance of each problem using CPLEX 8.1. We coded (*RPMP*) and (*RFLP*) in AMPL and solved the 50-node Euclidean dataset using the same computer described in §5.1. All CPLEX options were set to their default values, except that we imposed a maximum time limit of 1,800 seconds (30 minutes). We tested six values of α for each problem. P was set to 10 for the (*RPMP*) instances. The results are reported in Table 5. The columns labeled LR time and LR obj. indicate the CPU time and objective value returned by our Lagrangian relaxation algorithm (reproduced from Tables 3 and 4), while those labeled CPLEX time and CPLEX obj. indicate the corresponding values for CPLEX. The last two columns report the number of simplex iterations and branch-and-bound nodes as reported by CPLEX.

CPLEX required more time than our Lagrangian algorithm to solve 10 of the 12 problems tested, and for half of the problems, CPLEX could not prove optimality within the 30-minute limit, although CPLEX did find an optimal or near-optimal solution for all but one of the problems. CPLEX seemed to have an easier time solving the RFLP than the RPMP, solving 5 of the 6 RFLP problems without branching. (This is the result of CPLEX's preprocessing

Table 5 CPLEX Performance

Problem	α	LR time	CPLEX time	LR obj.	CPLEX obj.	# Simplex iter.	# BB nodes
RPMP	1.0	44.0	4.5	1,645	1,645	5,017	0
($P = 10$)	0.8	27.3	>1,800.0	1,689	1,689	118,568	5
	0.6	10.8	>1,800.0	1,732	1,732	192,730	9
	0.4	4.5	>1,800.0	1,776	1,776	98,591	0
	0.2	11.4	>1,800.0	1,819	1,819	121,438	2
	0.0	7.4	>1,800.0	1,863	1,863	307,822	117
RFLP	1.0	3.6	32.2	6,733	6,734	11,487	0
	0.8	2.9	79.0	6,214	6,214	39,009	0
	0.6	4.2	151.5	5,617	5,618	46,791	0
	0.4	6.5	101.1	4,866	4,867	38,104	0
	0.2	60.3	>1,800.0	3,561	4,079	47,498	0
	0.0	70.6	8.0	81	81	2,975	0

and cut-generation features; it does not mean that the LP relaxation has integer optimal solutions. For example, the RFLP with $\alpha = 0.6$ has an optimal objective value of 5,617, while its LP relaxation has an optimal objective value of 5,598.) This discussion suggests that CPLEX is not a reliable tool for solving the problems at hand, motivating the development of the Lagrangian relaxation algorithm presented above.

5.4. Trade-off Curves

We constructed the trade-off curve for the RFLP using the 49-node dataset as described in §3.3; the results are pictured in Figure 3. The horizontal axis plots the UFLP cost (Objective 1) while the vertical axis plots the expected failure cost (Objective 2). Each point on the curve represents a different solution; the optimal UFLP solution (found by solving (*RFLP*) with $\alpha = 1$) is the leftmost point on the curve. The steepness of the left part of the curve indicates that large improvements in reliability can be attained without large increases in UFLP cost. The 10 leftmost solutions on the curve are listed in Table 6, along with their relationship to the UFLP solution and the number of facilities open in the solution. Decision makers may be reluctant to undertake large increases in UFLP cost, but they may be willing, for example, to expend 7% more to reduce expected failure cost by 27%, as in Solution 4. The points at the right of the trade-off curve are unlikely to be of much interest, as they represent solutions in which nearly all of the facilities are open; they have extremely low failure costs but are very expensive to implement. In general, we find that the left portion of the trade-off curve is quite steep, indicating that substantial improvements in reliability can be attained with minimal increases in operational cost.

The trade-off curve in Figure 3 was constructed assuming all facilities are failable. Firms may be interested in knowing how the trade-off curve changes if some of the facilities in J are nonfailable. This may help firms make decisions about which contracts should be shored up or which facilities should be

Table 6 First 10 Solutions in Curve: RFLP

Solution	Obj. 1	Obj. 2	% Increase Obj. 1	% Decrease Obj. 2	# Locations
1	856,810	532,199	—	—	7
2	860,078	514,758	0.4	3.3	7
3	883,656	460,699	3.1	13.4	8
4	919,203	391,149	7.3	26.5	9
5	946,914	356,139	10.5	33.1	10
6	984,969	326,149	15.0	38.7	11
7	1,014,350	306,754	18.4	42.4	12
8	1,062,410	275,649	24.0	48.2	13
9	1,104,380	250,493	28.9	52.9	14
10	1,151,970	226,437	34.4	57.5	15

owned in-house, for example. Figure 4 contains two trade-off curves, one in which all facilities are failable (discussed earlier) and another in which 25 of the 49 facilities, chosen randomly, are designated as nonfailable. The inclusion of nonfailable facilities has the effect of shifting the trade-off curve favorably toward the origin. (In addition, problems with more nonfailable facilities generally produce tighter bounds and require less computation time.)

A natural question to ask is: How many nonfailable facilities need to be included in the set of potential facility sites to achieve a satisfactory level of reliability? Figure 5 addresses this question by plotting the decrease in objective function value as the number of nonfailable facilities increases, using the RFLP and the 49-node dataset with $\alpha = 0.8$ and $q = 0.10$. In the lower curve (marked "Last solution"), the nonfailable facilities were selected from the previous solution. This curve represents how the objective function might change if the firm could make any facilities that it likes nonfailable. In the upper curve (marked "Random"), the nonfailable facilities are chosen randomly. This represents the case in which the firm has no control over which facilities are nonfailable. Note that in both cases, the horizontal axis plots the number of nonfailable facilities that are in J; not all of these will necessarily be chosen in the solutions. From the chart it is apparent that only a few nonfailable DCs are necessary to make the solution as

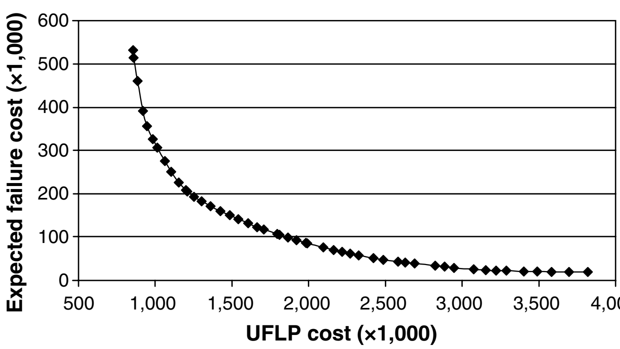

Figure 3 RFLP Trade-off Curve for 49-Node Dataset

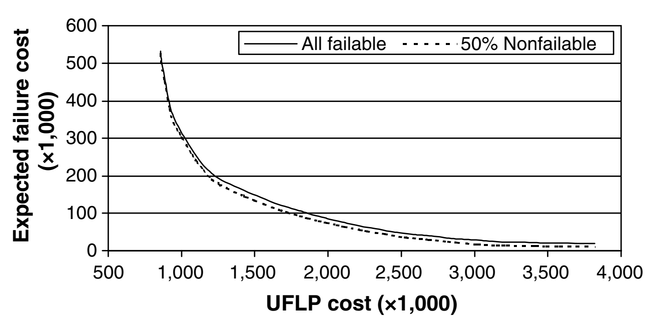

Figure 4 Shifting Trade-off Curve

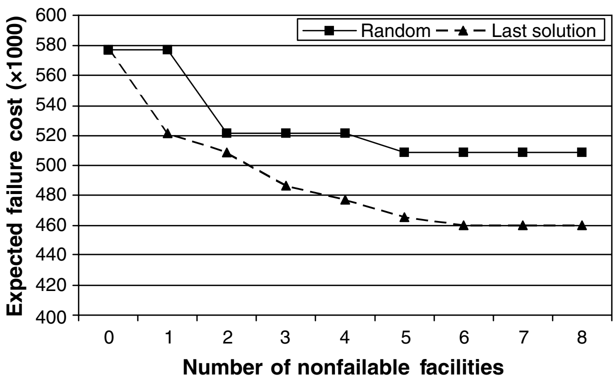

Figure 5 **Changing the Number of Nonfailing Facilities**

a whole substantially more reliable. For our client, the durable goods manufacturer, this means that the company needs to own only a few DCs in-house to make the system perform well when third-party distributors fail. It also allows the firm to threaten to cancel contracts with badly performing distributors, because the firm can credibly claim to be able to operate without the distributor, at least in the short term, while they establish a contract with a new distributor. This is an important issue in contract negotiation and enforcement.

5.5. Number of Levels

In this section we briefly explore the impact of changing the number of levels, m, as discussed in §4.3. We solved the RPMP with $P = 10$ and the RFLP using the 88-node dataset, $\alpha = 0.8$, and $q = 0.05$, testing various values of m. The results are presented in Table 7. (Note that the last entry for each problem corresponds to the case in which we do not use the modification suggested in §4.3. Because the emergency facility has been added to the problem, $m = 50$ for the RFLP and $m = 11$ for the RPMP as $|J|$ and P increased by 1 when the emergency facility was added.) The column marked Time/iter. gives the average number of CPU seconds spent on each Lagrangian iteration.

Table 7 **Sensitivity to** m

Problem	m	Root gap (%)	# Lag iter.	# BB nodes	CPU time	Time/iter.
RPMP	3	0.10	391	1	1.3	0.003
RPMP	5	0.10	592	1	2.7	0.005
RPMP	7	0.88	2,018	5	10.9	0.005
RPMP	9	1.24	7,185	19	44.4	0.006
RPMP	11	1.11	6,445	17	46.7	0.007
RFLP	3	0.07	120	1	0.5	0.005
RFLP	5	0.09	201	1	1.1	0.006
RFLP	8	0.98	4,102	11	31.5	0.008
RFLP	12	0.64	4,233	11	41.0	0.010
RFLP	20	0.66	4,279	11	57.4	0.013
RFLP	50	0.54	3,996	9	102.1	0.026

From the table it is apparent that using larger values of m generally results in larger root-node optimality gaps, more Lagrangian iterations, more branch-and-bound nodes, and more time spent on each iteration (due to loops of the form "for $r = 0, \ldots, m-1$"). It is worth pointing out that in all cases, regardless of the value of m, the algorithm returned the same set of facility locations (one set for the RPMP and one for the RFLP), indicating that the computational improvement came at no loss of solution accuracy, though of course we cannot prove that this will hold in general.

6. Conclusions

In this paper we presented two new models that incorporate reliability into classical facility location problems. These models arose from a realization that supply chains are vulnerable to disruptions of all sorts, and that facility location decisions can be critical in reducing the impact of these disruptions. We formulated reliability models based on the P-median problem and the uncapacitated fixed-charge location problem, called the RPMP and RFLP, respectively. Key to our formulation is the concept of "backup" assignments, which represent the facilities to which customers are assigned when closer facilities have failed. In both models, the expected transportation cost, taking into account the costs that result from facility failures, is included in the objective function. The trade-off of interest is between the operating cost (the traditional PMP or UFLP objective function) and the expected failure cost. In a subsequent paper, we will address the trade-off between operating cost and the maximum failure cost over all facilities in the solution. Both models are solved using Lagrangian relaxation, with promising results. We demonstrated empirically that the interesting portion of the trade-off curve is steep, indicating that reliability can be drastically improved without large increases in operating costs. This is a critical issue for decision makers who may be reluctant to expend greater sums *for sure* to hedge against *possible* failures in the future.

For large values of P in the RPMP, and for the RFLP, straightforward application of our algorithm yielded large gaps at the root node. We proposed a modification that entails assigning facilities only to a prespecified level m (we used $m = 5$). This modification tightens these bounds considerably with little or no loss of accuracy. In our computational tests, we found that the choice of m has a large impact on computational performance, but *no* impact on the solution found. For different values of m, the objective function for the solutions differed slightly because higher-level terms are excluded for smaller values of m. However, we found this difference to be less than 0.02% in all cases. This addresses an important question about the bounds produced by our algorithm. In particular, when a Lagrangian relaxation algorithm produces lower bounds that are loose, one always wants to know whether this is the *theoretical* lower bound (the maximum value of $z(\lambda)$ in (20), regardless of whether we can find it in practice) or simply a *practical* lower bound (the bound actually produced by a computer implementation) that might be improved by a different multiplier-updating method or different choices of algorithm parameters. Consider the last entry in Table 7. When we began testing, we assumed that the theoretical bound for this problem was 0.54% away from the optimal solution, or close to it. When $m = 3$, however, we get a lower bound that is only 0.07% from the upper bound, and as this upper bound is very close (within 0.02%, as discussed above) to the upper bound when all assignment levels are included in the objective function, we can be confident that the theoretical lower bound is no more than 0.1% from the optimal solution. This suggests that the size of the practical bounds is to some extent determined by our implementation of the multiplier updating routine, and not by the theoretical bound, and that we might tighten this bound even further by improving this routine or by choosing better initial multipliers. (This is especially important for the larger problems tested, which resulted in bounds significantly larger than 0.1% at the root node.)

We also note that our upper-bound heuristic and improvement routines are highly effective, yielding the optimal solution at the root node in 99 of 120 problems tested (83%), including all of the RFLP problems, often finding it within the first 100 iterations or so. This suggests that very good solutions can be found very quickly if a theoretical guarantee of optimality is not required.

The main drawback of our models is the assumption that failable facilities all have the same probability q of failing. This assumption is necessary to allow us to compute the probability that a customer is served by its level-r facility without explicitly knowing its lower-level assignments, only that there are r of them and that they are failable. Increasing the number of probabilities results in an exponential increase in the number of terms in the objective function because one term is required for each possible combination of the failure probabilities of the r lower-level assignments. We intend to study this issue in future research to find an objective function that can accommodate multiple failure probabilities.

Another limitation of our models is the assumption that the facilities are uncapacitated. While this is a common assumption made in facility location models, it may be unrealistic in practice. In the reliability context, customers of failed facilities can only be assigned to backup facilities if those facilities have enough excess capacity to accommodate the additional demand. This makes the capacitated case significant from a practical perspective; it also makes the capacitated model more complex than capacitated versions of classical models. In any case, the capacitated model is another topic worthy of future study.

Finally, we note that if decision makers are interested only in total expected cost, not in the trade-off between the PMP or UFLP objective and the expected failure cost, the two objectives can easily be replaced with a single objective representing the expected cost. For the RPMP, this simply means setting $\alpha = 0$. For the RFLP, one would add the fixed cost term to w_2 and then set $\alpha = 0$. In either case, the solution method remains the same. Some decision makers may prefer these formulations, as they address the common objective of minimizing long-run cost. We have chosen to formulate the problems as we did because the multiobjective framework provides greater flexibility; more importantly, it allows us to demonstrate, via trade-off curves, the large improvements in reliability that are possible with only small increases in the objectives under which firms have historically evaluated facility location decisions.

Acknowledgments
This research was supported by NSF Grants DMI-9812915 and DMI-0223277. This support is gratefully acknowledged. The authors would also like to thank the anonymous referees, whose comments improved the paper significantly.

References

Ball, M. O. 1979. Computing network reliability. *Oper. Res.* **27**(4) 823–838.

Ball, M. O., F. L. Lin. 1993. A reliability model applied to emergency service vehicle location. *Oper. Res.* **41**(1) 18–36.

Berman, O., D. Krass. 2001. Facility location problems with stochastic demands and congestion. Zvi Drezner, H. W. Hamacher, eds. *Facility Location: Applications and Theory*. Springer-Verlag, New York, 331–373.

Berman, O., R. C. Larson, S. S. Chiu. 1985. Optimal server location on a network operating as an $M/G/1$ queue. *Oper. Res.* **33**(4) 746–771.

Bienstock, D., E. F. Brickell, C. L. Monma. 1990. On the structure of minimum-weight k-connected spanning networks. *SIAM J. Discrete Math.* **3**(3) 320–329.

Bundschuh, M., D. Klabjan, D. L. Thurston. 2003. Modeling robust and reliable supply chains. *Optimization Online e-print*. www.optimization-online.org.

Church, R. L., C. ReVelle. 1974. The maximal covering location problem. *Papers Regional Sci. Assoc.* **32** 101–118.

Cohon, J. L. 1978. *Multiobjective Programming and Planning*. Academic Press, New York.

Colbourn, C. J. 1987. *The Combinatorics of Network Reliability*. Oxford University Press, New York.

Daskin, M. S. 1982. Application of an expected covering model to EMS system design. *Decision Sci.* **13** 416–439.

Daskin, M. S. 1983. A maximum expected covering location model: Formulation, properties and heuristic solution. *Transportation Sci.* **17**(1) 48–70.

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

Daskin, M. S., K. Hogan, C. ReVelle. 1988. Integration of multiple, excess, backup, and expected covering models. *Environment Planning B* **15**(1) 15–35.

Eiselt, H. A., M. Gendreau, G. Laporte. 1996. Optimal location of facilities on a network with an unreliable node or link. *Inform. Processing Lett.* **58**(2) 71–74.

Fisher, M. L. 1981. The Lagrangian relaxation method for solving integer programming problems. *Management Sci.* **27**(1) 1–18.

Fisher, M. L. 1985. An applications oriented guide to Lagrangian relaxation. *Interfaces* **15**(2) 10–21.

Fortz, B., M. Labbé. 2002. Polyhedral results for two-connected networks with bounded rings. *Math. Program.* **93**(1) 27–54.

Grötschel, M., C. L. Monma, M. Stoer. 1995. Polyhedral and computational investigations for designing communication networks with high survivability requirements. *Oper. Res.* **43**(6) 1012–1024.

Krass, D., O. Berman, M. Menezes. 2003. Reliability issues in median models. *EURO/INFORMS Joint Internat.* Istanbul, Turkey (July 6–10).

Larson, R. C. 1974. A hypercube queuing model for facility location and redistricting in urban emergency services. *Comput. Oper. Res.* **1** 67–95.

Larson, R. C. 1975. Approximating performance of urban emergency service systems. *Oper. Res.* **23**(5) 845–868.

Lynn, B. 2002. Unmade in America: The true cost of a global assembly line. *Harper's* **304**(1825) 33–41.

Martha, J., E. Vratimos. 2002. Creating a just-in-case supply chain for the inevitable next disaster. *Viewpoint* (Marsh & McLennan Companies' journal) 44–58.

Menezes, M., O. Berman, D. Krass. 2003a. The median problem and reliability of facilities. *INFORMS Annual Meeting*, Atlanta, GA (October 19–22).

Menezes, M., O. Berman, D. Krass. 2003b. MiniSum with imperfect information. *INFORMS Annual Meeting*, Atlanta, GA (October 19–22).

Monma, C. L., D. F. Shallcross. 1989. Methods for designing communications networks with certain 2-connected survivability constraints. *Oper. Res.* **37**(4) 531–541.

Monma, C. L., B. S. Munson, W. R. Pulleyblank. 1990. Minimum-weight 2-connected spanning networks. *Math. Programming* **46**(2) 153–171.

Navas, D. 2003. Automotive embraces volatility. *Supply Chain Systems* 12–16.

Pirkul, H. 1989. The uncapacitated facility location problem with primary and secondary facility requirements. *IIE Trans.* **21**(4) 337–348.

ReVelle, C., K. Hogan. 1989. The maximum availability location problem. *Transportation Sci.* **23**(3) 192–200.

Sheffi, Y. 2001. Supply chain management under the threat of international terrorism. *Internat. J. Logist. Management* **12**(2) 1–11.

Shier, D. R. 1991. *Network Reliability and Algebraic Structures*. Clarendon Press, Oxford, UK.

Shooman, M. L. 2002. *Reliability of Computer Systems and Networks: Fault Tolerance, Analysis, and Design*. John Wiley and Sons, New York.

Simchi-Levi, D., L. V. Snyder, M. Watson. 2002. Strategies for uncertain times. *Supply Chain Management Rev.* **6**(1) 11–12.

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

Vidal, C. J., M. Goetschalckx. 2000. Modeling the effect of uncertainties on global logistics systems. *J. Bus. Logist.* **21**(1) 95–120.

Weaver, J. R., R. L. Church. 1985. A median location model with nonclosest facility service. *Transportation Sci.* **19**(1) 58–74.