Joint Inventory-Location Problem under the Risk of Probabilistic Facility Disruptions

Qi Chen[a], Xiaopeng Li[b], Yanfeng Ouyang[b,1]

[a] Department of Automation, Tsinghua University, Beijing, China

[b] Department of Civil and Environmental Engineering, University of Illinois at Urbana-Champaign, Urbana, IL 61822

Abstract – This paper studies a reliable joint inventory-location problem that optimizes facility locations, customer allocations, and inventory management decisions when facilities are subject to disruption risks (e.g., due to natural or man-made hazards). When a facility fails, its customers may be reassigned to other operational facilities in order to avoid the high penalty costs associated with losing service. We propose an integer programming model that minimizes the sum of facility construction costs, expected inventory holding costs and expected customer costs under normal and failure scenarios. We develop a Lagrangian relaxation solution framework for this problem, including a polynomial-time exact algorithm for the relaxed nonlinear subproblems. Numerical experiment results show that this proposed model is capable of providing a near-optimum solution within a short computation time. Managerial insights on the optimal facility deployment, inventory control strategies, and the corresponding cost constitutions are drawn.

Keywords - joint inventory-location problem, facility location, disruption, Lagrangian relaxation

1 Introduction

Facility location problems have been intensively studied in the past few decades due to their wide applications in numerous contexts such as supply chain planning, public service provision, and transportation infrastructure deployment. A large number of problem variants have appeared since the original formulation in Weber's work in late 1950s (Weber, 1957). Daskin (1995) provides a thorough review of traditional discrete facility location models, and more recent studies focus on variants of these problems; e.g., Daskin et al. (2002), Shen et al. (2003), Snyder and Daskin (2005), Shu et al. (2005), Azad and Davoudpour (2008) and Cui et al. (2010). Various continuum approximation facility location models have also been proposed as alternatives to the traditional discrete models (Newell, 1971, 1973; Daganzo, 1984a, b; Ouyang and Daganzo, 2006; Ouyang, 2007, Li and Ouyang 2010a).

Many past studies (e.g., Drezner, 1995; Daskin and Owen 1998, 1999) focused on the uncapacitated fixed-charge location problem (UFL) that seeks the optimal number of

[1] Corresponding author. Tel: 217-333-9858; E-mail: yfouyang@illinois.edu.

facilities and their locations in a supply chain network to balance the trade-off between initial facility setup costs and day-to-day shipment costs. However, inventory costs were not typically considered in UFL. In many contexts where product safekeeping is expensive, the inventory holding cost may account for a significant portion of the total system cost. Applying UFL models to cases with significant inventory costs may yield suboptimal design and erroneous system cost estimation. Hence researchers proposed joint inventory-location models that optimize facility locations to minimize the summation of the inventory costs, the facility setup costs, and the customer transportation costs. Various solution algorithms such as Lagrangian relaxation (Daskin et al., 2002) and column generation (Shen et al., 2003) were proposed to solve the joint inventory-location models. Shu et al. (2005) further improved these algorithms by exploiting certain special structures in the models. Meta-heuristics algorithms have also been used to solve the joint inventory-location problem (e.g., Azad and Davoudpour, 2008).

Traditional facility location studies assume that a facility, once built, will remain functioning forever (or throughout its life cycle). However, many facilities are subject to potential operational disruptions from time to time. Such disruptions can cause severe damages to overall system efficiency and service quality. For example, when some facilities are not available, their customers either are forced to travel excessive distances so as to access more distant services, or entirely give up the service and suffer certain penalty. Snyder and Daskin (2005) proposed two reliable facility location model formulations (based on p-median and UFL models) to investigate the effect of probabilistic facility failures on the optimal facility deployment. Cui et al. (2010) extended these models to address site-dependent facility failure probabilities in both discrete and continuous modeling frameworks. Li and Ouyang (2010a) further improved the continuum approximation model so as to solve problems under complex facility failure patterns (such as those involving spatial correlation). These discrete and continuous reliable facility location modeling techniques have been adapted to solve traffic surveillance sensor location design problems (Li and Ouyang, 2010b, 2011).

Inventory management under supply chain disruption involves difficult nonlinear cost components, and such problems have been considered only very recently (e.g., Ross et al., 2008; Qi et al., 2009; Schmitt et al., 2010). In the reliable location design framework, some very recent studies tried to develop models to address the joint planning of inventory and facility location. For example, Qi et al. (2007) studied reliable delivery of final products to satisfy stochastic customer demand when the supply chain is subject to random yield at the facilities. Qi et al. (2010) further investigated the effects of facility disruptions at two supply chain echelons (e.g., supplier and retailers) on optimal retailer locations and customer allocations, while the facilities' disruption-recovery cycles are described by memoryless exponential distributions. Nevertheless, both studies assumed that a customer is assigned to a fixed retailer, and if this retailer is not available the customer loses service. In many realistic supply chain systems, if we allow customers to access backup services from other facilities (when their primary service facility has been disrupted), the supply chain system reliability and overall performance would be considerably improved.

Hence, we propose in this paper a nonlinear mixed-integer model to incorporate inventory costs and a more general customer assignment scheme into the reliable facility location design framework. This model can find the optimal facility location design and customer assignment strategy that minimize the expected total system cost across all possible operating scenarios (e.g., under probabilistic facility failures). We propose a customized Lagrangian relaxation approach that decomposes the model into a set of relatively easier subproblems. A polynomial-time algorithm is developed to solve each subproblem to its exact optimality despite the presence of nonlinear components. A number of case studies are conducted to test the proposed solution approach and draw insights on the optimal facility deployment.

The remainder of this paper is organized as follows. Section 2 introduces the notation and the model formulation. Section 3 proposes the Lagrangian relaxation solution approach and analyzes its major properties. Section 4 conducts numerical experiments to test the proposed approach and draw managerial insights. Section 5 concludes the paper and briefly discusses future research directions.

2 Model Formulation

2.1 Notation

We assume that a set of customers, **I**, are located at discrete locations, and each customer $i \in \mathbf{I}$ generates a constant demand λ_i. Facilities can be built anywhere among a set of candidate locations, **J**, to serve these customers. Opening a facility at $j \in \mathbf{J}$ incurs an initial setup cost that translates to an annual equivalent of $f_j > 0$. Once facilities are built, customers will visit nearby facilities for service. We assume that the annual transportation cost for customer i to visit location j is d_{ij}.

Each facility replenishes its inventory from time to time. We assume that the lead time for order delivery is negligible and therefore each facility orders exactly when its inventory is depleted. In this case, it has been proven that at the optimum, a facility at *j* shall place orders of a constant quantity Q_j periodically (Zipkin, 2000). Placing an order for a facility at $j \in \mathbf{J}$ incurs a fixed cost $b_j > 0$ and a variable cost $p_j > 0$ per unit of order. Carrying a unit of commodity in the inventory of that facility incurs a holding cost $h_j > 0$ per year.

We assume that each facility, once built, fails independently with an equal probability *q*. When a facility fails, it cannot provide any service and its original customers will be either diverted to other functioning facilities or subject to certain penalty. We assume that each customer is allowed to get service from a sequence of $R \le |\mathbf{J}|$ facilities.[2] Under this assumption, in the normal scenario (where no facilities fail), a customer is assigned to its level-1 facility. Whenever a customer's level-*r* facility fails (for any $r \le R-1$), it will be re-assigned to its level-(r+1) facility. When all its *R* assigned facilities have failed, the customer gives up service and suffers a penalty cost π per unit of its unmet demand. Note that due to independent failures, the probability for a customer to get service from its level-*r* facility is $(1-q)q^{r-1}$, i.e., the probability that its level-*r* facility is functioning while

[2] Such restriction can be caused by service compatibility, system capacity, service time requirement, or simply excessive transportation cost (Cui et al., 2010).

all lower-level facilities have failed. The probability for a customer to incur penalty is q^R, i.e., the probability that all of its R assigned facilities have failed.

The primal binary decision variables $Y=\{y_j\}_{j\in\mathbf{J}}$ determine the facility locations; i.e.,

$$y_j=\begin{cases}1, \text{if a facility is built at } j;\\ 0, \quad\quad\quad\quad\quad \text{otherwise.}\end{cases}$$

Given Y, the auxiliary binary variables $X=\{x_{ijr}\mid i\in\mathbf{I}, j\in\mathbf{J}, r=1,\cdots,R\}$ decide how facilities are assigned to the customers; i.e.,

$$x_{ijr}=\begin{cases}1, \text{if customer } i \text{ is assigned to facility } j \text{ at level } r;\\ 0, \quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad \text{otherwise.}\end{cases}$$

2.2 Formulation

The objective of this reliable joint inventory-location problem (RJIL) is to determine the optimal number of facilities and their locations that minimize the expected total cost (including the facility setup cost, the transportation cost and the inventory cost) across all possible facility failure scenarios. Given facility location decisions Y, the total facility initial setup cost is simply $\sum_{j\in\mathbf{J}} f_j y_j$, while the total expected customer cost consists of the expected penalty cost and the expected shipment cost as follows

$$\pi\sum_{i\in\mathbf{I}}\lambda_i q^R+\sum_{j\in\mathbf{J}}\sum_{i\in\mathbf{I}}\sum_{r=1}^{R}\lambda_i x_{ijr} d_{ij}(1-q)q^{r-1}. \tag{2.1}$$

Note that the total expected penalty cost $\pi\sum_{i\in\mathbf{I}}\lambda_i q^R$ is a constant and it can be omitted from the optimization model. The inventory cost consists of those related to ordering and holding. For a facility at j, its expected annual demand is $\sum_{i\in\mathbf{I}}\sum_{r=1}^{R}\lambda_i x_{ijr}(1-q)q^{r-1}$. Then its annual inventory cost is

$$\frac{\sum_{i\in\mathbf{I}}\sum_{r=1}^{R} b_j\lambda_i x_{ijr}(1-q)q^{r-1}}{Q_j}+\sum_{i\in\mathbf{I}}\sum_{r=1}^{R} p_j\lambda_i x_{ijr}(1-q)q^{r-1}+\frac{h_j Q_j}{2}. \tag{2.2}$$

For any given facility location and customer assignment, (2.2) forms an EOQ trade-off and the optimal ordering quantity is $Q_j^*=\left(\frac{2b_j}{h_j}\sum_{i\in\mathbf{I}}\sum_{r=1}^{R}\lambda_i x_{ijr}(1-q)q^{r-1}\right)^{\frac{1}{2}}$. Then the total expected inventory cost under the optimal ordering quantities is as follows:

$$\sum_{j\in\mathbf{J}}\left[\left(2b_j h_j\sum_{i\in\mathbf{I}}\sum_{r=1}^{R}\lambda_i x_{ijr}(1-q)q^{r-1}\right)^{\frac{1}{2}}+\sum_{i\in\mathbf{I}}\sum_{r=1}^{R}\lambda_i x_{ijr} p_j(1-q)q^{r-1}+f_j y_j\right]. \tag{2.3}$$

Summarizing the above, the RJIL problem can be expressed as follows,

$$(RJIL)\quad \min_{X,Y} F(X,Y) := \sum_{j\in\mathbf{J}} \left[\left(\sum_{i\in\mathbf{I}} \sum_{r=1}^{R} \alpha_{ijr} x_{ijr} \right)^{\frac{1}{2}} + \sum_{i\in\mathbf{I}} \sum_{r=1}^{R} \beta_{ijr} x_{ijr} + f_j y_j \right], \tag{2.4a}$$

subject to

$$\sum_{j\in\mathbf{J}} x_{ijr} = 1, \forall i \in \mathbf{I}, r \in \{1,2,...,R\} \tag{2.4b}$$

$$\sum_{r=1}^{R} x_{ijr} \le y_j, \forall i \in \mathbf{I}, j \in \mathbf{J} \tag{2.4c}$$

$$x_{ijr}, y_j \in \{0,1\}, \forall i \in \mathbf{I}, j \in \mathbf{J}, r \in \{1,2,...,R\} \tag{2.4d}$$

where $\alpha_{ijr} := 2b_j h_j \lambda_i (1-q)q^{r-1}$ and $\beta_{ijr} := \lambda_i (p_j + d_{ij})(1-q)q^{r-1}$. The objective function (2.4a) minimizes the total expected system cost (without the constant penalty costs). Constraints (2.4b) postulate that a customer is only assigned to one facility at each assignment level. Constraints (2.4c) ensure that a customer can only go to a location with a built facility, and that no customer goes to the same facility at two or more levels. Constraints (2.4d) define binary variables.

2.3 Remarks

The RJIL problem seeks to balance between the shipment cost (i.e., by spreading out customer demand across a large number of facilities) and the inventory and the facility setup costs (i.e., by pooling demand at a few facilities). In extreme cases where the transportation cost is relatively insignificant (compared with the inventory cost)[3] and the variable ordering cost is identical everywhere, i.e. $d_{ij} = 0, \forall i \in \mathbf{I}, j \in \mathbf{J}$ and $p_j = p, \forall j \in \mathbf{J}$, we have $\beta_{ijr} = \lambda_i p(1-q)q^{r-1}$ which is independent of *j*. Because the total demand is fixed, the term $\sum_{j\in\mathbf{J}} \sum_{i\in\mathbf{I}} \sum_{r=1}^{R} \beta_{ijr} x_{ijr}$ in (2.4a) becomes a constant and we can remove it from the objective function. An optimal solution to this special case has the following simple structure.

Proposition 1. *When* $d_{ij} = 0, \forall i \in \mathbf{I}, j \in \mathbf{J}$ *and* $p_j = p, \forall j \in \mathbf{J}$, *in an optimal solution to the RJIL problem, the following holds: (i) constraints (2.4c) are binding; i.e.,* $\sum_{r=1}^{R} x_{ijr} = y_j, \forall i \in \mathbf{I}, j \in \mathbf{J}$; *and (ii) for all* $j \in J$ *and* $r \in \{1,2,...,R\}$, *the value of* x_{ijr} *is identical across all* $i \in \mathbf{I}$, *i.e.,* $x_{ijr} = x_{i'jr}, \forall i, i' \in \mathbf{I}$.

Proof: See Appendix A.

In light of Proposition 1, when $d_{ij} = 0, \forall i \in \mathbf{I}, j \in \mathbf{J}$ *and* $p_j = p, \forall j \in \mathbf{J}$, we can denote the value of x_{ijr} as x_{jr}. Model (2.4) reduces to

[3] Such a special case may occur when the inventory is extremely costly to carry (e.g., high-valued luxuries) or the shipment is relatively cheap (e.g., electronic devices and digital materials).

$$\min_{X,Y} \sum_{j\in\mathbf{J}} \sum_{r=1}^{R} x_{jr} \left(\left(\sum_{i\in\mathbf{I}} \alpha_{ijr} \right)^{\frac{1}{2}} + f_j \right) \quad (2.5a)$$

subject to

$$\sum_{j\in\mathbf{J}} x_{jr} = 1, \forall r \in \{1,2,...,R\} \quad (2.5b)$$

$$x_{jr} \in \{0,1\}, \forall j \in \mathbf{J}, r \in \{1,2,...,R\} \quad (2.5c)$$

This is in the form of a typical assignment problem, which can be solved in $O(JR^2)$ time (Munkres, 1957).

Another extreme case occurs when the inventory cost is relatively insignificant, and our problem reduces to the reliable location models with equal facility disruption probabilities across all candidate locations (Snyder and Daskin, 2005). It has been shown that in this case, every customer shall always be assigned to its nearest functioning facility at each level.[4]

However, when the inventory cost is taken into consideration, the "nearest" assignment rule may no longer hold. Rather, assigning certain customs to a facility other than their nearest ones could actually reduce the system cost if the inventory cost saving exceeds the increased traveling cost. Daskin et al. (2002) demonstrated this fact for the deterministic inventory-location problem. Here, we provide a similar example for cases with facility disruptions and customer reassignments.

Suppose there are two identical facilities constructed at j = 1, 2, and two customers i = 1 and 2. The parameters (with suitable units) are: $R = 2$, $q = 0.1$, $f_1 = f_2 = 1000$, $p_1 = p_2 = 1$, $b_1 = b_2 = 1$, $h_1 = h_2 = 10$, $\pi_1 = \pi_2 = 1$, $\lambda_1 = 10$, $\lambda_2 = 1000$. The distances between the facilities and customers are given in Table 1.

Table 1: Distances between facilities and customers

Customer \ Facility	1	2
1	1	1.01
2	1	0.1

It is easy to verify by enumeration that the optimal customer assignment solution is to assign both customers to facility 2 at level 1 and to facility 1 at level 2, although facility 1 is closer to customer 1 than facility 2. This simple example shows that when the inventory cost is considered, the customers cannot be assigned merely based on their proximity to the facilities.

[4] It shall be noted that this assignment rule may no longer be valid if facility disruption probabilities vary across candidate locations; see Cui et al. (2010) for an example.

Therefore, when neither shipment costs nor inventory costs are negligible, the optimal solution does not have any simple structure. The next section proposes a solution approach that effectively solves such general problems.

3 The Lagrangian Relaxation Algorithm

Obviously, the nonlinear integer programming model (2.4) is NP-hard because the well-known p-median model is a special case (by setting $b_j = h_j = p_j = q = 0$), and therefore no known exact methods can solve RJIL efficiently. We propose a customized Lagrangian relaxation algorithm to find near-optimum solutions with optimality gaps.

We relax constraints (2.4b) and add them to objective (2.4a) with a set of Lagrangian multipliers $\mathbf{u} = \{u_{ir} \in \mathbb{R}\}$. This yields the following relaxed problem:

$$(LRJIL) \quad \begin{aligned} \Phi(\mathbf{u}) := \min_{X,Y} \sum_{j\in\mathbf{J}} &\left[\left(\sum_{i\in\mathbf{I}} \sum_{r=1}^{R} \alpha_{ir} x_{ijr} \right)^{\frac{1}{2}} + \sum_{i\in\mathbf{I}} \sum_{r=1}^{R} \left(\beta_{ijr} - u_{ir} \right) \cdot x_{ijr} + f_j y_j \right] \\ &+ \sum_{i\in\mathbf{I}} \sum_{r=1}^{R} u_{ir} \end{aligned}, \quad (3.1)$$

subject to (2.4c) and (2.4d).

For any given $\mathbf{u}$, $\Phi(\mathbf{u})$ is a lower bound of RJIL. Section 3.1 develops an exact solution approach to obtain $\Phi(\mathbf{u})$ for any $\mathbf{u}$. Based on the LRJIL results, Section 3.2 proposes an efficient algorithm to construct near-optimum solutions to the original RJIL problem. Section 3.3 briefly describes a standard subgradient method to update the Lagrangian multipliers.

3.1 Lower Bound

This section delineates how to solve $\Phi(\mathbf{u})$ for given multipliers $\mathbf{u}$. Note that the last term in (3.1), $\sum_{i\in\mathbf{I}} \sum_{r=1}^{R} u_{ir}$, is a constant and can be excluded from consideration. The relaxed problem can be decomposed into a set of subproblems across j, as follows.

$$\Phi_j(\mathbf{u}) := \min_{y_j, \{x_{ijr}\}_{\forall i,r}} \left(\sum_{i\in\mathbf{I}} \sum_{r=1}^{R} \alpha_{ir} x_{ijr} \right)^{\frac{1}{2}} + \sum_{i\in\mathbf{I}} \sum_{r=1}^{R} \left(\beta_{ijr} - u_{ir} \right) \cdot x_{ijr} + f_j y_j , \quad (3.2a)$$

subject to

$$\sum_{r=1}^{R} x_{ijr} \le y_j, \forall i \in \mathbf{I} \quad (3.2b)$$

$$x_{ijr}, y_j \in \{0,1\}, \forall i \in \mathbf{I}, j \in \mathbf{J}, r \in \{1,2,...,R\}. \quad (3.2c)$$

Obviously, $\Phi_j(\mathbf{u}) = 0$ when $y_j = 0$. Therefore we only need to solve the remaining case with $y_j = 1$, which is

$$\min_{\{x_{ijr}\}_{\forall i,r}} \left(\sum_{i\in\mathbf{I}} \sum_{r=1}^{R} \alpha_{ir} x_{ijr} \right)^{\frac{1}{2}} + \sum_{i\in\mathbf{I}} \sum_{r=1}^{R} \left(\beta_{ijr} - u_{ir} \right) \cdot x_{ijr} + f_j \quad (3.3a)$$

subject to

$$\sum_{r=1}^{R} x_{ijr} \le 1, \forall i \in \mathbf{I}, \tag{3.3b}$$

$$x_{ijr} \in \{0,1\}, \forall i \in \mathbf{I}, r \in \{1,2,...,R\}. \tag{3.3c}$$

In the rest of this section, we will focus on solving subproblem (3.3). For notation simplicity, we (i) omit subscript j in all relevant variables, (ii) omit the constant term f_j in (3.3a), and (iii) introduce slack variables $x_{i,R+1} \in \{0,1\}$ and further define $\bar{X} := \{x_{ir}\}_{\forall i,r}$, $\gamma_{ir} := \beta_{ir} - u_{ir}$, $\alpha_{i,R+1} := 0$, and $\gamma_{i,R+1} := 0$. Then subproblem (3.3) for any generic j is equivalent to the following:

$$\bar{\Phi}(\mathbf{u}) := \min_{\bar{X}} \left(\sum_{i\in\mathbf{I}} \sum_{r=1}^{R+1} \alpha_{ir} x_{ir} \right)^{\frac{1}{2}} + \sum_{i\in\mathbf{I}} \sum_{r=1}^{R+1} \gamma_{ir} x_{ir} \tag{3.4a}$$

subject to

$$\sum_{r=1}^{R+1} x_{ir} = 1, \forall i \in \mathbf{I} \tag{3.4b}$$

$$x_{ir} \in \{0,1\}, \forall i \in \mathbf{I}, r \in \{1,2,...,R+1\}. \tag{3.4c}$$

The deterministic version (i.e., when R=1 and q=0) of the subproblem (3.4) has been studied by Shen et al. (2002) and Daskin et al. (2003). However, due to constraints (3.4b), existing algorithms are no longer applicable to the general reliable version of this subproblem. Hence, we propose a customized polynomial-time algorithm to solve subproblem (3.4).

For any solution $\bar{X}$, the marginal contribution of setting $x_{ir} = 1$ can be denoted as $M_{ir}(w_i) := \sqrt{w_i + \alpha_{ir}} - \sqrt{w_i} + \gamma_{ir}, \forall i \in \mathbf{I}, r \in \{1,2,...,R+1\}$, where $w_i := \sum_{k\in\mathbf{I}\setminus\{i\}} \sum_{r=1}^{R+1} \alpha_{kr} x_{kr}$. Let $\mathbf{R}^i$ denote a maximal subset of $\{1,2,...,R+1\}$ whose elements have distinct $(\alpha_{ir}, \gamma_{ir})$ values; i.e., $(\alpha_{ir}, \gamma_{ir}) \ne (\alpha_{ir'}, \gamma_{ir'})$, $\forall r \ne r' \in \mathbf{R}^i$, and for any $r \in \{1,2,...,R+1\} \setminus \mathbf{R}^i$, there exists $r' \in \mathbf{R}^i$, such that $(\alpha_{ir}, \gamma_{ir}) = (\alpha_{ir'}, \gamma_{ir'})$. Then we define the following set:

$$N_i = \left\{ r \in \mathbf{R}^i \mid M_{ir}(0) < M_{ir'}(0) \text{ or } \gamma_{ir} < \gamma_{ir'}, \forall r' \in \mathbf{R}^i \setminus \{r\} \right\}.$$

Note that for any $r' \in \{1, 2, \dots, R+1\} \setminus N_i$, there exists $r \in N_i$ that satisfies $M_{ir'}(w_i) \ge M_{ir}(w_i), \forall w_i \in [0, +\infty)$. This implies that for an optimal solution $\bar{X}$, if we know w_i, then the level r with $x_{ir} = 1$ can be any element in the following set

$$\rho_i(w_i) := \{ r \in N_i \mid M_{ir'}(w_i) \ge M_{ir}(w_i), \forall r' \in N_i \}, \forall i \in \mathbf{I}.$$

Since $\forall r \ne r' \in N_i$, $(\gamma_{ir} - \gamma_{ir'})\left(M_{ir}(0) - M_{ir'}(0)\right) < 0$, continuous functions $M_{ir}(w_i)$ and $M_{ir'}(w_i)$ always intersect at

$$\bar{w}^i_{rr'} := \frac{(\alpha_{ir} - \alpha_{ir'})^2}{4(\gamma_{ir} - \gamma_{ir'})^2} + \frac{(\gamma_{ir} - \gamma_{ir'})^2}{4} - \frac{\alpha_{ir} + \alpha_{ir'}}{2} > 0.$$

This allows us to sort the elements of N_i into an ordered sequence $r(i,1), r(i,2)\cdots r(i,|N_i|)$ such that $M_{i,r(i,k)}(0) < M_{i,r(i,k+1)}(0)$, $\alpha_{i,r(i,k)} < \alpha_{i,r(i,k+1)}$ and $\gamma_{i,r(i,k)} > \gamma_{i,r(i,k+1)}, \forall 1 \le k \le |N_i| - 1$. We define a sequence of interval thresholds, $\{w_k^{i-}\}$ and $\{w_k^{i+}\}, \forall 1 \le k \le |N_i|$, as follows:

$$w_1^{i-} := 0, \quad w_{|N_i|}^{i+} := +\infty,$$

$$w_k^{i-} := \max_{k'=1,\cdots,k-1} \overline{w}^i_{r(i,k),r(i,k')}, \forall 2 \le k \le |N_i| \text{ and}$$

$$w_k^{i+} := \min_{k'=k+1,\cdots,|N_i|} \overline{w}^i_{r(i,k),r(i,k')}, \forall k = 1,\cdots,|N_i| - 1.$$

Then intervals $\left[w_k^{i-}, w_k^{i+}\right], \forall 1 \le k \le |N_i|$ form a non-overlapping partition of $[0,+\infty)$ [5] from left to right (as shown in Figure 1); i.e., $w_k^{i+} \le w_{(k+1)}^{i-}, \forall 1 \le k \le |N_i| - 1$, and $\bigcup_{1 \le k \le |N_i|} \left[w_k^{i-}, w_k^{i+}\right] = \left[0,+\infty\right]$.

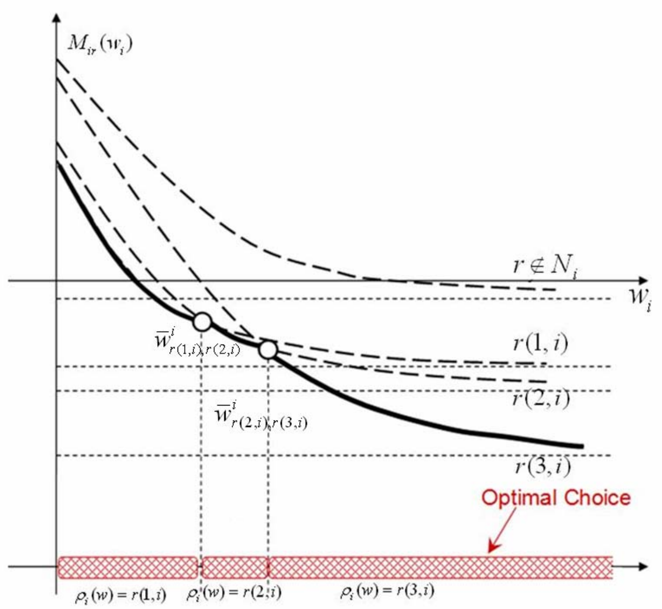


Figure 1: The interval partition over inventory cost level axis.

It turns out that the assignment levels in $\rho_i(w_i)$ and intervals $\left[w_k^{i-}, w_k^{i+}\right], \forall 1 \le k \le |N_i|$ have the following relationship.

---

[5] For notation simplicity, we have used $\left[\bullet,+\infty\right]$ and $\left[\bullet,+\infty\right)$ interchangeably. Also, note that $\left[w_k^{i-}, w_k^{i+}\right]$ is an empty set if $w_k^{i-} > w_k^{i+}$.

Proposition 2. *For all* $i \in \mathbf{I}$ *and* $w_i \geq 0$, $\rho_i(w_i) = \left\{ r(i,k) \mid w_i \in \left[w_k^{i-}, w_k^{i+} \right] \right\}$.

***Proof*:** See Appendix B.

According to Proposition 2, an optimal solution $\bar{X}$ satisfies $w_i \in \left[w_{k(i)}^{i-}, w_{k(i)}^{i+} \right]$ and $x_{ir} = 1$, $\forall i \in \mathbf{I}$, if and only if $r = r_{k(i)}^{i}$ for some $k(i) \in \left\{ 0, 1, \cdots, |N_i| \right\}$. This implies that if we define $\varpi_k^{i-} := w_k^{i-} + \alpha_{i,r_k^i}$, $\varpi_k^{i+} := w_k^{i+} + \alpha_{i,r_k^i}$, then $w := \sum_{i \in \mathbf{I}} \sum_{r=1}^{R+1} \alpha_{ir} x_{ir} \in \left[\varpi_{k(i)}^{i-}, \varpi_{k(i)}^{i+} \right]$, $\forall i \in \mathbf{I}$. We note that since $w_k^{i+} \leq w_{k+1}^{i-}$ and $\alpha_{i,r_k^i} < \alpha_{i,r_{k+1}^i}$, $\forall 1 \leq k \leq |N_i| - 1$, intervals $\left[\varpi_k^{i-}, \varpi_k^{i+} \right]$, $\forall 1 \leq k \leq |N_i|$, are mutually disjoint, and thus w can be only contained in one unique (i.e., the $k(i)^{th}$) interval among $\left[\varpi_k^{i-}, \varpi_k^{i+} \right], \forall 1 \leq k \leq |N_i|$. Hence, if w is contained in an intersection $\bigcap_{\forall i \in \mathbf{I}} \left[\varpi_{k'(i)}^{i-}, \varpi_{k'(i)}^{i+} \right]$ for some combination $\{k'(i)\}_{\forall i \in \mathbf{I}}$, then it is certain that $k(i) = k'(i), \forall i \in \mathbf{I}$, and $\bar{X}$ can be determined accordingly, i.e., $x_{ir} = 1$ if and only if $r = r_{k'(i)}^{i}$, $\forall i \in \mathbf{I}$. Figure 2 illustrates the non-empty intersections $\bigcap_{\forall i \in \mathbf{I}} \left[\varpi_{k'(i)}^{i-}, \varpi_{k'(i)}^{i+} \right]$ for all possible combinations $\{k'(i)\}_{\forall i \in \mathbf{I}}$. Obviously there are only a polynomial number of such intersections. Hence we can efficiently enumerate all these intersections to obtain a set of candidate solutions, among which the one with the minimum objective (3.4a) gives the exact optimal solution. This idea is described in the following algorithm.

Step A1: Compute $N_i, \forall i \in \mathbf{I}$ and $\varpi_k^{i-}, \varpi_k^{i+}$, $\forall k = 1, \cdots, |N_i|, i \in \mathbf{I}$; Initialize $\bar{X} = \{x_{ir}\}$ where $x_{ir} = 0, \forall i \in \mathbf{I}, r \in \{1, \cdots, R+1\}$.

Step A2: Sort pairs $\mathbf{W} := \left\{ (\varpi_k^{i-}, 0) \right\}_{\forall i \in \mathbf{I}, k=1,\cdots,|N_i|} \bigcup \left\{ (\varpi_k^{i+}, 1) \right\}_{\forall i \in \mathbf{I}, k=1,\cdots,|N_i|}$ into $\{(s_1, c_1), (s_2, c_2), \cdots, (s_{|\mathbf{W}|}, c_{|\mathbf{W}|})\}$ such that $s_1 \leq s_2 \leq \cdots \leq s_{|\mathbf{W}|}$;

Step A3.0: Initialize index $l := 1$ and candidate solution set $\mathbf{X} := \varnothing$;

Step A3.1: Repeat $l := l + 1$ until $c_l = 0$ and $c_{l+1} = 1$ when $l < |\mathbf{W}|$; or go to Step **A4** when $l = |\mathbf{W}|$.

Step A3.2: For all $i \in \mathbf{I}$, find $k'(i) \in \{1, \cdots, |N_i|\}$, s.t. $[s_l, s_{l+1}) \subseteq \left[\varpi_{k'(i)}^{i-}, \varpi_{k'(i)}^{i+} \right]$. If $k'(i)$ does not exist, $l := l + 2$ and go to Step A3.1; otherwise set $x_{i, r(i, k'(i))} = 1$ and $x_{ir} = 0, \forall r \in N_i \setminus \{ r(i, k'(i)) \}$.

Step A3.3: Compute $w = \sum_{i \in \mathbf{I}} \sum_{r=1}^{R+1} \alpha_{ir} x_{ir}^{*}$ for the current $\bar{X}$. If $w \in [s_l, s_{l+1})$, then $\mathbf{X} := \mathbf{X} \bigcup \{\bar{X}\}$, $l := l + 2$ and go to Step A3.1;

Step A4: Return the optimal solution $\bar{X} := \arg\min_{X \in \mathbf{X}} \bar{\Phi}(X)$.

It can be easily verified that Algorithm A1-A4 has a polynomial time complexity, $O(IR^2 + IR \log(IR))$. Recall that

$$\Phi_j(\mathbf{u}) = \min\{0, \bar{\Phi}(\mathbf{u}) + f_j\}. \tag{3.5}$$

If $\overline{\Phi}(\mathbf{u})+f_j>0$, we set $y_j=0$ and $x_{ijr}=0,\forall i\in\mathbf{I},r\in\{1,2,...,R\}$. Otherwise set $y_j=1$ and $X=\overline{X}$. Finally, the optimal objective value of the LRJIL problem (3.1) is

$$\Phi(\mathbf{u})=\sum_{j\in\mathbf{J}}\Phi_j(\mathbf{u})+\sum_{i\in\mathbf{I}}\sum_{r=1}^{R}u_{ir}\,. \tag{3.6}$$

This is a lower bound to the original RJIL problem.

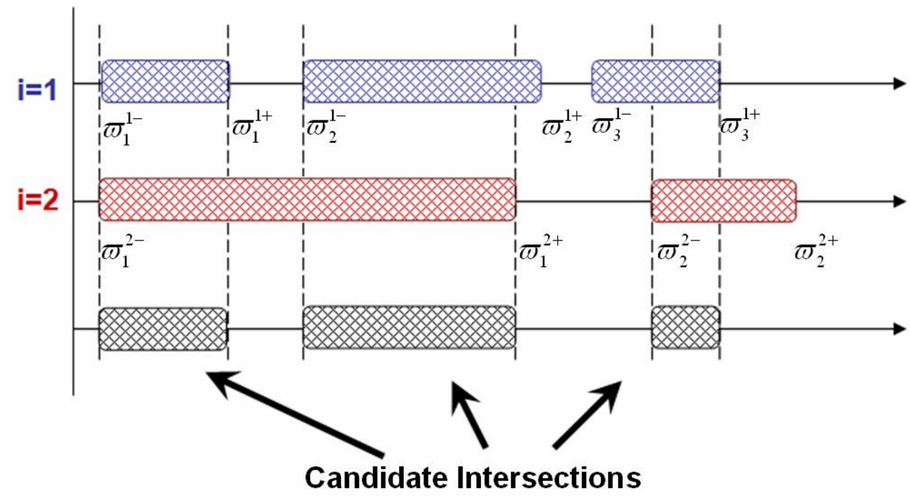


Figure 2: An illustration of candidate optimal solutions.

3.2 Feasible Solution and Upper Bound

A solution (*X*, *Y*) to the LRJIL problem, where $X:=\{x_{ijr}\}$ and $Y:=\{y_j\}$, may be infeasible to the original problem as it may violate the assignment constraints (2.4b). There are two types of possible violations posed by assignments *X*: (i) customer *i* is assigned to more than one facility at level *r* (i.e., $\sum_{j\in\mathbf{J}}x_{ijr}>1$), or (ii) it is not assigned to any facility at level *r* (i.e., $\sum_{j\in\mathbf{J}}x_{ijr}=0$). We propose two simple heuristics to obtain a feasible solution (and an upper bound) to RJIL based on the solution to LRJIL.

The basic idea of the first heuristic is to inspect for violations across all customers and assignment levels, and then iteratively update (*X*, *Y*) until it satisfies (2.4b) and all other constraints. We will first iteratively correct type (i) violations. For each customer *i* and assignment level *r* that satisfy $\sum_{j\in\mathbf{J}}x_{ijr}>1$, there are more than one elements in $\{x_{ijr}:\forall j\}$ that are equal to one. We can modify the solution by keeping exactly one of these elements to be one but setting all others to be zero. All other elements of X (i.e., those regarding other customers or other assignment levels) remain unchanged. This way, constraints (2.4c) are still satisfied because this modification does not increase the value of the left hand side. Further, we could update $Y:=\left\{\min\left(\sum_{i\in\mathbf{I}}\sum_{r=1}^{R}x_{ijr},1\right)\right\}_{\forall j}$ to remove facilities that are no longer used by any customers. As such, we have constructed a new solution that satisfies

$\sum_{j\in\mathbf{J}} x_{ijr} = 1$. Note that there are multiple such new solutions due to the multiple ways to set one element in $\{x_{ijr} : \forall j\}$ to be one. We evaluate all these solutions and choose the best one that minimizes the system objective (2.4a).

After this process, we shall have $\sum_{j\in\mathbf{J}} x_{ijr} \le 1$ for all i, r. Only type (ii) violations may remain. For each customer i and level r having $\sum_{j\in\mathbf{J}} x_{ijr} = 0$, we modify solution ($X$, Y) in a similar yet slightly different way. We set exact one element in $\{x_{ijr} : \forall j\}$ to one, while keeping all other elements in X unchanged. We also update $Y := \left\{\min\left(\sum_{i\in\mathbf{I}} \sum_{r=1}^{R} x_{ijr}, 1\right)\right\}_{\forall j}$ based on the modified X. Note that this new solution (X, Y) will satisfy $\sum_{j\in\mathbf{J}} x_{ijr} = 1$ but may also cause violations to (2.4c). However, among all $|\mathbf{J}|$ possible new solutions, there are at least one that does not violate (2.4c) because $|\mathbf{J}| \ge R$. We only evaluate those solutions that satisfy (2.4c), and pick the one with the best objective value (2.4a). After iterating this algorithm for all i, r, constraints (2.4b) shall hold. Finally, we let $Y := \left\{\min\left(\sum_{i\in\mathbf{I}} \sum_{r=1}^{R} x_{ijr}, 1\right)\right\}_{\forall j}$ (this will also eliminate any possible violations to constraints (2.4c)), and return (X, Y) as the feasible solution.

The second heuristic is much simpler. We fix the selected facility locations according to the LRJIL solution Y, and then update X by reassigning customers purely based on their distances to these facilities; i.e., customers are assigned to their nearest facility at the first level, the second nearest facility at the second level, and so forth. Note again that this assignment rule is generally not optimal in light of the discussion in Section 2.3. Instead, we use it as a heuristic approach. After iteratively updating the assignment strategies for all customers, we will obtain a feasible solution set (X, Y) to the original problem.

The above heuristics will each yield a feasible solution to RJIL. We will pick the one with the smaller objective value and use it to update the upper bound, if possible.

3.3 Multiplier Update

Since $\Phi(\mathbf{u})$ is a lower bound of RJIL for any given $\mathbf{u}$, we seek the optimal $\mathbf{u}$ that provides the tightest lower bound; i.e.

$$\max_{\mathbf{u}} \Phi(\mathbf{u}) . \tag{3.7}$$

Based on the lower bound and the upper bound solution approaches for a subproblem, we use the subgradient algorithm to update Lagrangian multipliers $\mathbf{u}$ as follows.

Step U1: Set initial multiplier values $\mathbf{u}^0 = \{u_{ir}^0 = 0\}$, step size parameter $0 < \tau^0 \le 2$, and iteration index k=0. Set the best known feasible objective of LRJIL $z^{UB} = +\infty$ (since no feasible solution is known at the very beginning).

Step U2: Solve subproblem $\Phi(\mathbf{u}^k)$ and obtain its optimal solution $X^k = \{x_{ijr}^k\}$ and $Y^k = \{y_j^k\}$. If $\Phi(\mathbf{u}^k)$ does not improve in K consecutive iterations (where K is a predefined number), $\tau^k = \tau^k / 2$.

Step U3: Adapt X^k and Y^k to a feasible solution in the way described in Section 3.2, and update $z^{UB} = \min\{z^{UB}, F(X^k, Y^k)\}$.

Step U4: Calculate the step size, $t^k := \dfrac{\tau^k (z^{UB} - \Phi(\mathbf{u}^k))}{\sum_{i=1}^{M} \sum_{r=1}^{R} (1 - \sum_{j=1}^{N} x_{ijr})^2}$, and update the multipliers accordingly, $u_{ir}^{k+1} = u_{ir}^k + t^k (1 - \sum_{j=1}^{N} x_{ijr})$

Step U5: Terminate the algorithm if (i) $z^{UB} - \Phi(\mathbf{u}^k)$ is smaller than a specified tolerance ε, (ii) τ^k is smaller than its minimum value Γ, or (iii) k exceeds a maximum iteration number K^m. Otherwise k=k+1, and go to Step U2.

4 Numerical Experiments

This section conducts three sets of numerical experiments to test the proposed model and its solution approach. We also draw managerial insights on how different problem settings affect the optimal facility location design and customer assignments. All the datasets are from Snyder and Daskin (2005): a 49-node set consisting of Washington D.C. and 48 continental state capital cities; an 88-node set consisting of the union of the 49-node set and the set of 50 largest cities in the United States; a 150-node set consisting of the 150 largest cities in the United States. Each city generates a customer demand proportional to its population and also serves as a candidate facility location. The fixed facility set-up costs of the 49-node and 88-node datasets are based on the local median house prices, while those of the 150-node dataset are identical across locations. The shipment cost d_{ij} between any two cities is proportional to the great-circle distance by a factor s.

The model and solution approach are implemented in C++ program on a PC with 2.00GHz CPU and 2GB RAM. In the Lagrangian relaxation algorithm, we set $\Gamma = 10^{-10}$, $\varepsilon = 1\%$, K=150, K^m= 5000. The results for the 49-node cases are summarized in Table 2. We have run 56 instances with b_j=1000, π=100, p_j=5, and a range of other parameters. The holding cost h_j is equal to a constant value of 10 for the first 28 instances, and it equals $10^{-3} f_j$ for the latter 28 instances. We see that all these instances can be solved to very tight optimality gaps (less than 1% for all instances) in a short time, which shows that our proposed algorithm can efficiently obtain near-optimum solutions. The average time per iteration increases almost linearly with R but decreases with s. This is probably because the pair-wise comparisons among indices in N_i in algorithm A1-A4 constitute the major computational burden in solving the subproblem, and the size of N_i in general increases with R and decreases with s.

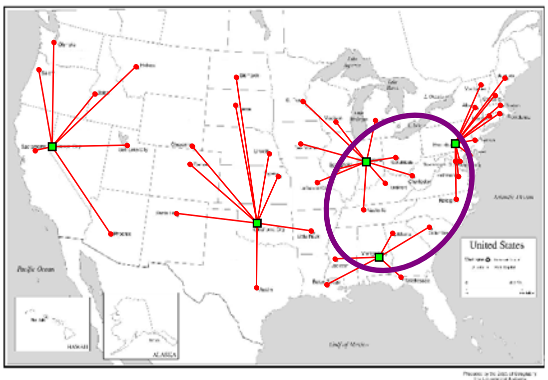
The total cost increases dramatically with q, due to the enormous additional cost incurred by customer reassignments. On the other hand, the total costs when R>1 are considerably lower than those with R=1, implying the significant benefit from providing back-up services. When R is small, the optimal number of facilities decreases with q, which is consistent with the conclusion in Qi et al. (2010). However, we also find that when R is large, the optimal number of facilities increases with q, which suggests that at a higher failure probability, additional facilities can provide better redundancy for reliable service quality against facility failures. This is because when our customers can be reassigned to more back-up facilities, the marginal penalty cost saving from one additional facility can better offset the extra infrastructure investment, thus making redundancy preferable. All cost components except the penalty cost are relatively insensitive to parameter changes, except that the penalty cost increases dramatically with q and decreases significantly with R. This cost distribution is similar to that under heterogeneous h_j.

We see that the optimal number of facilities is greater when s is larger (which means that the shipment cost gets more weight compared to the inventory cost). When the shipment cost is dominating, more facilities shall be deployed to reduce the average customer traveling distance. On the contrary, as discussed in Section 2.3, when the inventory cost is dominating, customer demand tends to be pooled to fewer facilities, and a customer may no longer be assigned to its nearest operating facility. For example, when s=0.05, q=0.1 and R=2, the customers in Montgomery are assigned to Oklahoma at the second level. Interestingly, Indianapolis is actually much closer (825.4 km from Montgomery) than Oklahoma's (1091.7 km away from Montgomery) but it is not chosen to serve Montgomery at any level. Among all of our 112 experiment cases, there are 13 cases in which at least one customer is assigned to a farther facility. Nevertheless, we find that in those 13 cases, the LR solution objective value is very close to that from the shortest distance heuristic (i.e., the second heuristic in Section 3.2), and the difference is on the same order of magnitude as the optimality gap.

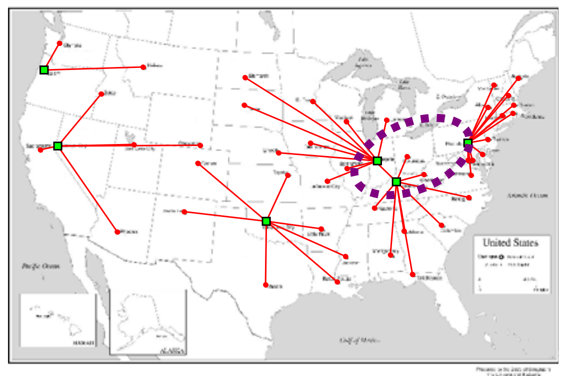
Table 2: Numerical results for the 49-node dataset.

No.	R	s	q	Opt. gap (%)	Solution time (sec)	Time per Iter. (sec)	No. of facilities	Total cost	Cost Components (%)				
									Inventory	Fixed set-up	Fixed order	Shipment	Others (Penalty)
1	3	0.05	10%	0.339	28.9	0.0299	5	114614	17.547	27.536	6.78	47.922	0.216
2	3	0.05	30%	0.93	33.9	0.0306	7	137169	15.331	30.933	6.568	42.306	4.863
3	3	0.05	50%	0.599	26.1	0.0292	6	163628	11.417	22.875	4.811	42.024	18.873
4	3	0.1	10%	0.666	20.7	0.0246	8	158322	13.982	35.788	6.188	43.886	0.156
5	3	0.1	30%	0.9	10.1	0.0226	9	188316	11.84	31.298	5.458	47.861	3.542
6	3	0.1	50%	0.968	12.9	0.0224	10	219246	9.607	29.93	4.678	41.7	14.085
7	5	0.05	10%	0.228	45.1	0.046	5	114642	17.557	27.529	6.782	48.13	0.002
8	5	0.05	30%	0.971	51.6	0.0458	7	137277	15.615	31.695	6.638	45.614	0.437
9	5	0.05	50%	0.997	33.8	0.0449	7	162852	12.866	25.588	5.518	51.288	4.741
10	5	0.1	10%	0.997	29.8	0.0372	8	158097	13.994	35.51	6.18	44.315	0.002
11	5	0.1	30%	0.964	30.2	0.0381	9	191129	11.895	30.838	5.447	51.506	0.314
12	5	0.1	50%	0.755	22.3	0.0374	11	227728	10.225	31.186	4.97	50.229	3.39
13	2	0.05	10%	0.04	16.5	0.022	5	114960	17.363	27.453	6.726	46.309	2.149
14	4	0.05	10%	0.975	24.0	0.0338	5	114627	17.558	27.533	6.783	48.105	0.022
15	6	0.05	10%	0.674	55.6	0.0553	6	116394	17.897	32.888	7.284	41.93	0
16	3	0.15	10%	0.965	18.5	0.022	9	191878	11.853	33.146	5.422	49.45	0.129
17	3	0.2	10%	0.934	5.6	0.0208	14	216925	11.637	43.517	5.948	38.784	0.114
18	1	0.05	10%	0.087	15.2	0.015	4	121679	14.584	20.669	5.448	38.995	20.304
19	1	0.05	30%	0.205	10.0	0.0147	4	156509	9.26	16.069	3.735	23.58	47.355
20	1	0.05	50%	0.854	9.4	0.0157	3	191086	5.451	8.023	2.218	19.664	64.644
21	1	0.1	10%	0.296	6.6	0.0128	6	158436	12.078	25.203	5.061	42.066	15.593
22	1	0.1	30%	0.782	9.5	0.0128	5	188885	7.998	18.519	3.42	30.824	39.238
23	1	0.1	50%	0.091	8.8	0.0132	5	216404	5.374	16.09	2.52	18.934	57.081
24	2	0.05	30%	0.254	16.8	0.0213	5	139039	13.301	22.541	5.217	42.95	15.992
25	2	0.05	50%	0.894	13.0	0.0219	5	169929	9.324	18.484	3.872	31.974	36.346
26	2	0.1	10%	0.163	15.1	0.0187	8	157214	13.963	35.709	6.184	42.572	1.571
27	2	0.1	30%	0.774	15.6	0.0184	8	184039	11.208	29.374	5.1	42.236	12.081
28	2	0.1	50%	0.482	12.0	0.0178	8	213317	8.339	25.343	3.996	33.369	28.954
29	3	0.05	10%	0.19	17.9	0.0282	5	111153	16.536	28.393	5.434	49.414	0.222
30	3	0.05	30%	0.99	53.9	0.0337	7	133381	14.108	31.241	5.097	44.552	5.001
31	3	0.05	50%	0.921	23.6	0.0283	7	159592	10.813	26.154	4.04	39.642	19.35
32	3	0.1	10%	0.903	23.1	0.0267	8	154537	13.198	36.328	5.213	45.102	0.16
33	3	0.1	30%	0.77	14.6	0.0255	9	183921	10.95	31.372	4.415	49.636	3.627
34	3	0.1	50%	0.501	17.8	0.0254	10	213984	8.829	30.376	3.778	42.585	14.432
35	5	0.05	10%	0.38	37.3	0.0504	5	111179	16.546	28.387	5.436	49.629	0.002
36	5	0.05	30%	0.935	37.6	0.049	7	132927	14.45	31.348	5.18	48.571	0.452
37	5	0.05	50%	0.988	50.7	0.0452	8	158375	12.107	29.436	4.551	49.03	4.875
38	5	0.1	10%	0.913	32.2	0.0418	8	154674	13.197	36.296	5.21	45.296	0.002
39	5	0.1	30%	0.777	26.3	0.0374	9	186758	11	30.896	4.402	53.382	0.321
40	5	0.1	50%	0.986	18.0	0.0385	11	223796	9.296	31.225	3.949	52.08	3.45
41	2	0.05	10%	0.239	21.4	0.0229	5	111517	16.357	28.301	5.391	47.736	2.215
42	4	0.05	10%	0.963	26.8	0.0356	5	111164	16.547	28.391	5.436	49.604	0.022
43	6	0.05	10%	0.422	64.1	0.0544	6	112127	16.836	33.034	5.819	44.31	0
44	3	0.15	10%	0.7	11.0	0.0224	10	186683	11.45	36.993	4.84	46.584	0.132
45	3	0.2	10%	0.868	8.2	0.0229	13	212366	10.625	41.099	4.814	43.346	0.116
46	1	0.05	10%	0.061	10.5	0.0149	4	118626	13.673	21.201	4.301	39.999	20.826
47	1	0.05	30%	0.152	8.7	0.0151	4	153816	8.547	16.351	2.925	23.993	48.184
48	1	0.05	50%	0.898	4.9	0.016	3	188552	4.852	8.13	1.576	19.929	65.513
49	1	0.1	10%	0.395	10.1	0.0132	6	155193	11.214	24.93	4.05	43.887	15.919
50	1	0.1	30%	0.789	6.1	0.0133	5	186447	7.449	18.761	2.811	31.227	39.752
51	1	0.1	50%	0.063	21.4	0.0139	5	214250	4.894	15.482	2.011	19.959	57.655
52	2	0.05	30%	0.696	14.9	0.0231	6	135984	12.65	25.775	4.384	40.839	16.351
53	2	0.05	50%	0.956	15.7	0.023	5	167756	8.461	17.943	2.939	33.84	36.817
54	2	0.1	10%	0.217	17.7	0.0196	7	153745	12.605	30.232	4.651	50.905	1.607
55	2	0.1	30%	0.683	14.5	0.019	8	180496	10.447	29.951	4.219	43.065	12.319
56	2	0.1	50%	0.067	17.5	0.0205	8	209709	7.66	25.316	3.242	34.331	29.452

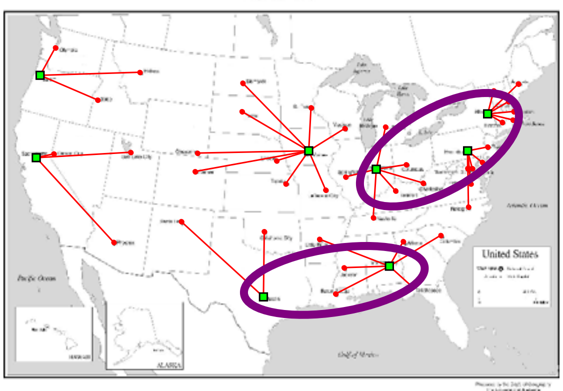
Figures 3 shows the optimal facility deployments and the corresponding level-1 customer assignments under q = 10% and q = 50%, respectively. In both Figures 3(a) and 3(b), R=3 and s=0.05**.** In Figure 3(a), five facilities are built in Sacramento, Oklahoma City, Indianapolis, Montgomery and Harrisburg, respectively. In Figure 3(b), the facility built in Montgomery moves to Frankfort and one more facility is built in Salem. We see that again, when R is large, the number of facilities increases as the failure probability increases. More interestingly, we see that facilities tend to cluster as failure probability increases (as highlighted in Figure 3(b). This clustering trend is more salient when s increases to 0.1, as shown in Figures 3(c) and 3(d). Intuitively, more clustered facilities can better back up each other and thus mitigate transportation cost increase in failure scenarios.



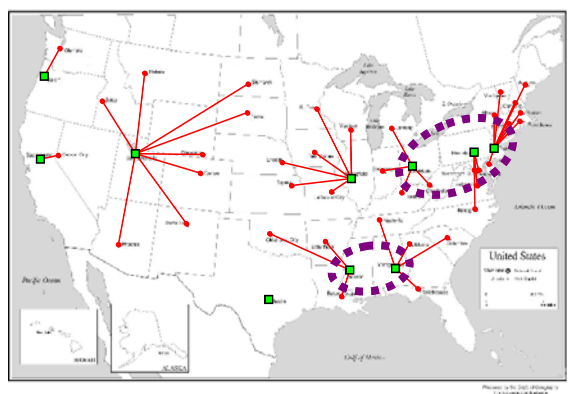
(a) q=10%, R=3 and s=0.05



(b) q=50%, R=3 and s=0.05



(c) q=10%, R=3 and s=0.1



(d) q=50%, R=3 and s=0.1

Figure 3: Facility location and customer assignment.

The numerical results for the 88-node and 150-node datasets are summarized in Tables 3 and 4 respectively. The results remain consistent with those from the 49-node dataset. Despite the increased problem sizes, the proposed solution approach can still solve most of these instances to less than 1% optimality gap within 30 minutes.

Table 3: Numerical results for the 88-node dataset.

No.	*R*	*s*	*q*	Opt. gap (%)	Solution time (sec)	Time per Iter. (sec)	No. of facilities	Total cost	Cost Components (%)				
									Inventory	Fixed set-up	Fixed order	Shipment	Others (Penalty)
1	3	0.05	10%	0.79	84.5	0.1178	6	149300	20.538	22.029	5.536	51.596	0.3
2	3	0.05	30%	0.916	81.3	0.1273	8	176204	17.785	24.324	5.405	45.615	6.871
3	3	0.05	50%	0.526	108.6	0.1208	8	222010	12.854	19.179	4.018	38.702	25.247
4	3	0.1	10%	0.802	61.1	0.1027	10	209612	15.93	27.56	5.244	51.052	0.214
5	3	0.1	30%	0.975	65.6	0.097	11	245679	13.388	24.617	4.509	52.558	4.928
6	3	0.1	50%	0.843	49.2	0.1059	13	293315	10.705	25.328	4.017	40.841	19.109
7	5	0.05	10%	0.984	183.9	0.1847	6	149248	20.563	22.037	5.541	51.856	0.003
8	5	0.05	30%	0.948	174.1	0.2388	9	173345	18.817	28.019	5.915	46.62	0.629
9	5	0.05	50%	0.878	188.3	0.2299	10	207233	15.56	25.614	5.08	46.984	6.762
10	5	0.1	10%	0.953	96.4	0.2037	10	209696	15.937	27.549	5.245	51.267	0.002
11	5	0.1	30%	0.888	90.0	0.1685	12	246432	13.851	26.51	4.776	54.42	0.442
12	5	0.1	50%	0.861	73.4	0.1916	14	291527	11.847	26.886	4.396	52.065	4.807
13	2	0.05	10%	0.687	39.5	0.0838	6	150457	20.218	21.86	5.466	49.475	2.98
14	4	0.05	10%	0.987	121.9	0.1569	6	149242	20.562	22.038	5.541	51.829	0.03
15	6	0.05	10%	0.898	278.5	0.2618	6	149250	20.563	22.037	5.541	51.859	0
16	3	0.15	10%	0.664	80.1	0.1123	12	258271	13.458	28.35	4.785	53.233	0.174
17	3	0.2	10%	0.95	75.1	0.1383	14	301808	12.193	33.319	4.771	49.568	0.149
18	1	0.05	10%	0.284	28.6	0.0657	5	170312	16.041	15.389	4.193	38.048	26.329
19	1	0.05	30%	0.861	38.7	0.0676	4	240377	8.886	8.874	2.357	23.92	55.963
20	1	0.05	50%	0.704	16.4	0.0667	4	307865	5.166	6.659	1.525	13.826	72.825
21	1	0.1	10%	0.105	37.3	0.0569	9	223575	13.316	22.534	4.291	39.804	20.056
22	1	0.1	30%	0.92	42.0	0.0584	7	287999	7.991	13.014	2.542	29.744	46.709
23	1	0.1	50%	0.069	33.7	0.0606	5	344269	4.802	7.613	1.546	20.914	65.124
24	2	0.05	30%	0.998	83.5	0.1041	8	189960	15.539	22.594	4.799	35.823	21.245
25	2	0.05	50%	0.952	41.4	0.0835	8	247971	10.129	17.308	3.348	24.007	45.207
26	2	0.1	10%	0.483	60.2	0.0756	10	209968	15.655	26.314	5.084	50.812	2.136
27	2	0.1	30%	0.857	41.8	0.0746	10	248462	12.313	21.931	4.101	45.412	16.243
28	2	0.1	50%	0.914	64.4	0.0789	10	303068	8.599	17.979	3.051	33.381	36.989

Table 4: Numerical results for the 150-node dataset.

No.	*R*	*s*	*q*	Opt. gap (%)	Solution time (sec)	Time per Iter. (sec)	No. of facilities	Total cost	Cost Components (%)				
									Inventory	Fixed set-up	Fixed order	Shipment	Others (Penalty)
1	3	0.05	10%	0.765	761.9	0.5339	5	198511	19.668	20.654	5.025	54.36	0.293
2	3	0.05	30%	0.067	1139.1	0.7223	7	237699	17.213	25.663	5.302	45.212	6.61
3	3	0.05	50%	0.018	955.6	0.5961	8	298936	12.828	23.751	4.311	34.775	24.335
4	3	0.1	10%	0.287	868.0	0.4542	10	281702	15.742	32.304	5.423	46.325	0.207
5	3	0.1	30%	0.745	716.0	0.4239	9	328802	13.069	24.635	4.458	53.059	4.779
6	3	0.1	50%	0.745	691.8	0.538	10	392232	10.216	23.201	3.724	44.313	18.547
7	5	0.05	10%	0.9	1670.2	0.81	5	198750	19.656	20.629	5.015	54.697	0.003
8	5	0.05	30%	0.279	3302.3	1.1364	7	235585	17.757	25.893	5.435	50.315	0.6
9	5	0.05	50%	0.863	1923.8	0.9841	9	285987	14.952	28.323	5.095	45.271	6.359
10	5	0.1	10%	0.664	1086.3	0.8305	10	281865	15.746	32.285	5.423	46.544	0.002
11	5	0.1	30%	0.011	972.9	0.6766	10	329287	13.588	27.635	4.773	53.575	0.429
12	5	0.1	50%	0.522	1877.8	1.0306	12	394653	11.496	28.126	4.353	51.416	4.608
13	2	0.05	10%	0.586	550.5	0.3641	5	199735	19.387	20.527	4.964	52.209	2.914
14	4	0.05	10%	0.867	865.7	0.6408	6	198836	20.228	25.649	5.595	48.498	0.029
15	6	0.05	10%	0.166	1560.4	1.0022	6	197230	20.409	25.858	5.656	48.077	0
16	3	0.15	10%	0.244	472.7	0.3306	13	333998	14.044	36.228	5.341	44.212	0.174
17	3	0.2	10%	0.027	456.9	0.342	14	380105	12.483	34.464	4.835	48.064	0.153
18	1	0.05	10%	0.961	155.3	0.2057	4	220665	15.653	14.048	3.785	40.14	26.373
19	1	0.05	30%	0.612	139.9	0.212	4	309583	8.959	10.013	2.38	22.253	56.395
20	1	0.05	50%	0.003	348.9	0.2308	4	398113	5.217	7.787	1.563	12.343	73.091
21	1	0.1	10%	0	398.7	0.1785	7	296797	12.654	20.553	3.83	43.354	19.608
22	1	0.1	30%	0.479	249.9	0.1936	5	375986	7.628	10.905	2.211	32.821	46.435
23	1	0.1	50%	0.087	218.5	0.1988	4	447399	4.644	6.929	1.392	21.997	65.039

24	2	0.05	30%	0.957	600.0	0.4292	6	253163	14.843	20.145	4.384	39.939	20.689
25	2	0.05	50%	0.974	801.6	0.461	7	330706	9.908	18.445	3.309	24.344	43.994
26	2	0.1	10%	0.013	441.1	0.3057	9	281025	15.374	28.823	5.124	48.608	2.071
27	2	0.1	30%	0.004	518.5	0.3554	9	332352	12.232	24.372	4.265	43.372	15.759
28	2	0.1	50%	0.019	448.8	0.3886	8	400216	8.466	17.74	3.014	34.426	36.353

5 Conclusion and Future Research

This paper proposes a reliable joint inventory-shipment facility location model that incorporates a general customer assignment mechanism and the inventory ordering and holding costs into the reliable facility location design framework. This model determines the optimal number of facilities and their locations, the corresponding customer assignments and inventory management policies that minimize the expected inventory, customer and facility set-up costs across all possible facility disruption scenarios. We formulated a compact nonlinear integer program and developed a customized solution approach to efficiently obtain near-optimum solutions and the corresponding optimality gaps. Numerical results show that the proposed approach is able to obtain solutions with very tight optimality gaps in a short time under various problem settings. Managerial insights about the problem are drawn from these results. For example, we have found that customer demand tend to be pooled together for service by only a few facilities when the inventory cost is dominating, while it will be spread to more facilities to reduce the shipment when the transportation cost is dominating. When the facility failure probability increases, the expected total system cost and the number of constructed facilities both increase, and the facility locations tend to cluster together.

This work can be further extended in several directions. The presence of lead time or backorders may affect supply chain structure and facility location design. This shall be addressed in future studies. In the real world, due to spatial heterogeneity and interdependence of facility failure hazards, facility failure probabilities may present complex patterns such as site-dependence and spatial correlation. It would be interesting to study how different facility failure patterns affect facility location design. As the problem scale increases, the discrete model may become computationally intractable. It might be appealing to develop alternative approximation models to tackle large-scale reliable joint inventory location problems.

Acknowledgments

This research was supported in part by the National Science Foundation through Grant CMMI-0748067 and EFRI-RESIN-0835982. The first author conducted this research when he was visiting the University of Illinois. The helpful comments from the two anonymous reviewers are gratefully acknowledged.

References

Azad, N., Davoudpour, H. 2008. A hybrid Tabu-SA algorithm for location-inventory model with considering capacity levels and uncertain demands. *Journal of Information and Computing Science* 3 (4), 290-304.

Bertsekas, D. P. 1999. Nonlinear Programming. Athena Scientific, Belmont, Mass.

Cui, T., Ouyang, Y., Shen, Z.M., 2010. Reliable facility location under the risk of disruptions. *Operations Research* 58 (4), 998-1011.

Daganzo, C. 1984a. The distance traveled to visit N points with a maximum of C stops per point: a manual tour-building strategy and case study. *Transportation Science* 18 (4), 331–350.

Daganzo, C. 1984b. The length of tours in zones of different shapes. *Transportation Research Part B* 18 (2), 135–146.

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

Daskin, M. S. 2008. What you should know about location modeling. *Naval Research Logistics*. 55 (4), 283-294.

Daskin, M. S., Coullard, C., Shen, Z.J. 2002. An inventory-location model: Formulation, solution algorithm and computational results. *Annals of Operations Research* 110 (1),83-106.

Daskin, M. S., Owen, S. H. 1999. Location models in transportation. in: Handbook of Transportation Science, Kluwer Academic, Norwell, MA, 311–360.

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

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

Li, X. and Ouyang, Y. 2010a. A continuum approximation approach to reliable facility location design under correlated probabilistic disruptions. *Transportation Research Part B* 44 (4), 535-548.

Li, X. and Ouyang, Y. 2010b. Reliable traffic sensor deployment under probabilistic disruptions and generalized surveillance effectiveness measures. *Operations Research*. Minor revision under review.

Li, X. and Ouyang, Y. 2011. Reliable sensor deployment for network traffic surveillance. *Transportation Research Part B* 45(1), 218-231.

Munkres, J. 1957. Algorithms for the assignment and transportation problems. *Journal of the Society for Industrial and Applied Mathematics*, 5 (1), 32–38,

Newell, G. 1971. Dispatching policies for a transportation route. *Transportation Science* 5(1), 91–105.

Newell, G. 1973. Scheduling, location, transportation and continuum mechanics: some simple approximations to optimization problems. *SIAM Journal on Applied Mathematics* 25(3), 346 – 360.

Ouyang, Y. 2007. Design of vehicle routing zones for large-scale distribution systems. *Transportation Research Part B* 41 (10), 1079 – 1093.

Ouyang, Y., Daganzo, C. 2006. Discretization and validation of the continuum approximation scheme for terminal system design. *Transportation Science* 40 (1), 89 – 98.

Owen, S. H., Daskin, M. S. 1998. Strategic facility location: A review. *European Journal of Operations Research* 111 (3), 423–447.

Qi, L. and Shen, Z.J. 2007. A supply chain design model with unreliable supply. *Naval Research Logistics* 54 (8), 829-844.

Qi, L., Shen, Z.J., and Snyder, L. V. 2009. A continuous-review inventory model with disruptions at both supplier and retailer. *Production and Operations Management* 18(5), 516-532.

Qi, L., Shen, Z.J., and Snyder, L. V. 2010. The effect of supply disruptions on supply chain design decisions. *Transportation Science* 44 (2), 274-289.

Ross, A.M., Rong, Y. and Snyder, L.V. 2008. Supply disruptions with time-dependent parameters. *Computers and Operations Research* 35 (11), 3504 – 3529.

Schmitt, A., Snyder, L.V. and Shen, Z.J. 2010. Inventory systems with stochastic demand and supply: Properties and approximations. *Journal of Operational Research* 206 (2), 271 – 508.

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

Shu, J., Teo, C.P., Shen, Z.J. 2005. Stochastic transportation-inventory network design problem. *Operations Research* 53 (1), 48-60.

Snyder, L. V. and Daskin, M. S. 2005. Reliability Models for Facility Location: The Expected Failure Case. *Transportation Science* 39 (3), 400-416.

Weber, A. 1957. Theory of the Location of Industries. University of Chicago Press, IL.

Zipkin, P. H. 2000. Foundations of Inventory Management. Irwin, Burr Ridge, IL.

Appendix A. Proof for Proposition 1.

To prove Proposition 1, we first show that the following lemma holds.

Lemma 1. *For any positive real numbers A, A’, C, C’, and B, if B<A≤A’ and C≥C’, then*

$$\left(AC\right)^{\frac{1}{2}}+\left(A'C'\right)^{\frac{1}{2}}-\left[\left(A-B\right)C\right]^{\frac{1}{2}}-\left[\left(A'+B\right)C'\right]^{\frac{1}{2}}>0\,.$$

Proof. Simple algebraic manipulation shows that the above inequality is equivalent to

$$(AC)^{\frac{1}{2}} - \left[(A-B)C\right]^{\frac{1}{2}} > \left[(A'+B)C'\right]^{\frac{1}{2}} - (A'C')^{\frac{1}{2}}$$
$$\Leftrightarrow \frac{BC}{(AC)^{\frac{1}{2}} + \left[(A-B)C\right]^{\frac{1}{2}}} > \frac{BC'}{\left[(A'+B)C'\right]^{\frac{1}{2}} + (A'C')^{\frac{1}{2}}}$$
$$\Leftrightarrow (A/C)^{\frac{1}{2}} + \left[(A-B)/C\right]^{\frac{1}{2}} < (A'/C')^{\frac{1}{2}} + \left[(A'+B)/C'\right]^{\frac{1}{2}}$$

The last inequality is obviously true. □

Now we are ready to prove Proposition 1.

Proposition 1. *When* $d_{ij} = 0, \forall i \in \mathbf{I}, j \in \mathbf{J}$ *and* $p_j = p, \forall j \in \mathbf{J}$, *in an optimal solution to the RJIL problem, the following holds: (i) constraints (2.4c) are binding; i.e.,* $\sum_{r=1}^{R} x_{ijr} = y_j, \forall i \in \mathbf{I}, j \in \mathbf{J}$; *and (ii) for all* $j \in J$ *and* $r \in \{1, 2, ..., R\}$, *the value of* x_{ijr} *is identical across all* $i \in \mathbf{I}$, *i.e.,* $x_{ijr} = x_{i'jr}, \forall i, i' \in \mathbf{I}$.

Proof. Statement (i) essentially claims that exactly *R* facilities are constructed in an optimal solution. Since each customer is assigned to *R* distinct facilities, the number of constructed facilities shall be no less than *R*. We only need to show that the number of constructed facilities is no greater than *R*. We will prove this by contradiction.

Assume that the optimal solution is $X = \{x_{ijr}\}$ and $Y = \{y_j\}$, where $\sum_{j=1}^{J} y_j = R' > R$. Define $A_j := \sum_{i=1}^{I} \sum_{r=1}^{R} 2\lambda_i (1-q) q^{r-1} x_{ijr}$ and $C_j := h_j k_j$, $\forall j \in \mathbf{J}$. Then $\sqrt{A_j C_j}$ represents the inventory cost at facility *j*. Without losing generality, we assume that the *R'* facilities are constructed at locations 1, 2, …, *R*', and $C_1 \le C_2 \le ... \le C_{R'}$. For any $1 < j < j' \le R'$ such that $C_j < C_{j'}$, if $A_j < A_{j'}$, we can strictly decrease the objective value by swapping the values of x_{ijr} and $x_{ij'r}, \forall i \in \mathbf{I}$, $r = 1, 2, \ldots, R$, which contradicts the optimality of the solution. Hence, we must have $A_1 \ge A_2 \ge ... \ge A_{R'}$. For a customer *i* that is assigned to facility *R'* at some level *r* (i.e., $x_{iR'r} = 1$), there must exist a facility $j \in \{1, 2, \ldots, R'-1\}$ such that $\sum_{r'=1}^{R} x_{ijr'} = 0$. Then swapping the values of $x_{iR'r}$ and x_{ijr} (i.e., by letting $x_{iR'r} = 0$ and $x_{ijr} = 1$) will decrease the objective value by

$$\left(A_{R'} C_{R'}\right)^{\frac{1}{2}} + \left(A_j C_j\right)^{\frac{1}{2}} - \left(\left(A_{R'} - 2\lambda_i (1-q) q^{r-1}\right) C_{R'}\right)^{\frac{1}{2}} - \left(\left(A_j + 2\lambda_i (1-q) q^{r-1}\right) C_j\right)^{\frac{1}{2}} > 0.$$

The above inequality holds from Lemma 1 if we define $B = 2\lambda_i (1-q) q^{r-1}$ and notice $B < A_{R'} \le A_j$ *and* $C_{R'} \ge C_j$. This contradicts the optimality of the solution. This proves (i).

Again, we assume that in an optimal solution, the *R* facility locations are 1, 2, …, *R* such that $C_1 \le C_2 \le ... \le C_R$ and $A_1 \ge A_2 \ge ... \ge A_R$. We will prove by contradiction a stronger claim that each customer is assigned to facility *r* at level *r*, $\forall r = 1, 2, \ldots, R$. Assume that there exists a customer $i \in \mathbf{I}$ that does not satisfy this condition. Let r be the smallest facility location index that satisfy $x_{irr} = 0$. Then there exists $r', r'' \in \{r+1, \ldots R\}$ that satisfy $x_{ir'r} = 1$ and $x_{irr''} = 1$. Then by setting $x_{ir'r} = x_{irr''} = 0$ and $x_{ir'r''} = x_{irr} = 1$, we obtain a feasible solution that decreases the objective by

$$\left(A_{r'}C_{r'}\right)^{\frac{1}{2}} + \left(A_r C_r\right)^{\frac{1}{2}} - \left(\left(A_{r'} - \lambda_i(1-q)(q^{r-1} - q^{r''-1})\right)C_{r'}\right)^{\frac{1}{2}}$$
$$- \left(\left(A_r + \lambda_i(1-q)(q^{r-1} - q^{r''-1})\right)C_r\right)^{\frac{1}{2}} > 0,$$

which contradicts with the optimality assumption. The inequality again comes from Lemma 1. Hence, each customer shall be assigned to facility *r* at level *r* in this optimal solution, which implies (ii). This completes the proof. □

Appendix B. Proof for Proposition 2.

Proposition 2. *For all* $i \in \mathbf{I}$ *and* $w_i \geq 0$, $\rho_i(w_i) = \left\{ r(i,k) \mid w_i \in \left[w_k^{i-}, w_k^{i+} \right] \right\}$.

***Proof*:** By the definition of $\rho_i(w_i)$, the proposition holds if we can prove the following statement: for any $k \in N_i$, $M_{i,r(i,k^*)}(w_i) \geq M_{i,r(i,k)}(w_i),\ \forall k^* \neq k \in \{1,2,\ldots,|N_i|\}$ if and only if $w_i \in [w_k^{i-}, w_k^{i+}]$.

We first prove the sufficiency. Given $w_i \in [w_k^{i-}, w_k^{i+}]$, $\forall k^* > k$, $w_i \leq w_k^{i+} = \min_{k'=k+1,\ldots,|N_i|} \overline{w}^i_{r(i,k),r(i,k')} \leq \overline{w}^i_{r(i,k),r(i,k^*)}$. Since $M_{i,r(i,k)}(0) < M_{i,r(i,k^*)}(0)$ and there is only one intersection of continuous functions $M_{i,r(i,k)}(w_i)$ and $M_{i,r(i,k^*)}(w_i)$, then $M_{i,r(i,k^*)}(w_i) \geq M_{i,r(i,k)}(w_i)$. We can prove in a similar way that the same conclusion holds when $k^* < k$.

Then we prove the necessity. If $w_i \notin [w_k^{i-}, w_k^{i+}]$, then $\exists k^* \in N_i$ such that $w_i \in [w_{k^*}^{i-}, w_{k^*}^{i+}]$. If $k^* < k$, $w_i \leq w_{k^*}^{i+} = \min_{k'=k^*+1,\ldots,|N_i|} \overline{w}^i_{r(i,k),r(i,k')} \leq \overline{w}^i_{r(i,k),r(i,k^*)}$. Since $w_i \notin [w_k^{i-}, w_k^{i+}]$, we know that $w_i, w_{k^*}^{i+}$ and $\overline{w}^i_{r(i,k),r(i,k^*)}$ cannot be all identical, and thus $w_i < \overline{w}^i_{r(i,k),r(i,k^*)}$. Since $M_{i,r(i,k)}(0) > M_{i,r(i,k^*)}(0)$ and there is only one intersection of continuous functions $M_{i,r(i,k)}(w_i)$ and $M_{i,r(i,k^*)}(w_i)$, then $M_{i,r(i,k^*)}(w_i) < M_{i,r(i,k)}(w_i)$. We can prove in a similar way that the same conclusion holds when $k^* > k$. This completes the proof. □