International Regional Science Review
34(3) 339-367

DOI: 10.1177/0160017610386483
http://irsr.sagepub.com

Constructs for Multilevel Closest Assignment in Location Modeling

Ting L. Lei[1] and Richard L. Church[1]

Abstract

In the classic *p*-median problem, it is assumed that each point of demand will be served by his or her closest located facility. The *p*-median problem can be thought of as a "single-level" allocation and location problem, as all demand at a specific location is assigned as a whole unit to the closest facility. In some service protocols, demand assignment has been defined as "multilevel" where each point of demand may be served a certain percentage of the time by the closest facility, a certain percentage of the time by the second closest facility, and so on. This article deals with the case in which there is a need for "explicit" closest assignment (ECA) constraints. The authors review past location modeling work that involves single-level ECA constraints as well as specific constraint constructs that have been proposed to ensure single-level closest assignment. They then show how each of the earlier proposed ECA constructs can be generalized for the "multilevel" case. Finally, the authors provide computational experience using these generalized ECA constructs for a novel multilevel facility interdiction problem introduced in this article. Altogether, this article proposes both a new set of constraint structures that can be used in location models involving multilevel assignment as well as a new facility interdiction model that can be used to optimize worst case levels of facility disruption.

[1] Department of Geography, University of California, Santa Barbara, CA, USA

Corresponding Author:
Ting L. Lei, Department of Geography, University of California, Santa Barbara, CA 93106, USA
Email: tinglei@geog.ucsb.edu

Keywords
facility location, single and multilevel allocation, closest assignment, spatial analysis

Introduction

Over the past four decades, a number of location models have been developed to optimize service and logistics systems. One of the key issues is the form in which demands for service and product are allocated and handled across a system of facilities. In many industrial settings, this allocation is based on the minimization of transport costs, represented as a classic transportation problem or a multicommodity flow problem. For public service systems, however, it is often assumed that users or clients will be served by their closest facility, which is called closest assignment. Closest assignment (CA) may be an inherent feature of a problem. For example, the classic p-median problem involves the location of a set of p facilities across a network to serve a number of demands positioned at the vertices or nodes of a network. Each node has an assigned weight that represents some form of need, such as the number of people being served. The objective of this problem is to place the facilities so that the total weighted distance of serving all demand is minimized. Since it is assumed each facility has the capacity to serve whatever is assigned to it, the optimal solution to the p-median problem will involve the assignment of each demand to their closest located facility, whether it is required or not.

In a "public fiat" system (Wagner and Falkson 1975), users can be arbitrarily assigned to facilities to achieve the optimal system objective. This means that certain users may be assigned to far away or inconvenient facilities. It is clear that such a system policy is only possible when services are delivered to users or when the institutional environment warrants such an assignment policy. However, in most location problems for public facilities, customers can choose which facilities to use. In a user-oriented system, users will not sacrifice their own self-interest for system objectives by going to faraway facilities and the model should allocate users to the closest or most preferred available facility.

It is unfortunate that in many circumstances, CA will not occur without special constraints to enforce it, like the "public fiat" system of Wagner and Falkson. This issue was first encountered by Rojeski and ReVelle (1970) in the development of the budget-constrained median problem. Without special constraints, Rojeski and ReVelle found that users were assigned to "cost-convenient" facilities, rather than "customer-chosen" facilities. Since the development of the budget constrained median problem, there have been many models that have been proposed that require specific constraints to force CA to facilities. One of the latest examples of this is the r-interdiction median problem (RIM) of Church, Scaparra, and Middleton (2004), where the objective is to maximally disrupt an existing system of p facilities by striking or destroying r of the p facilities. This problem represents the point of view of an interdictor or attacker, so the objective is to disrupt the system the most with a limited amount of resources. Specifically, the objective is to maximize total weighted distance by removing r facilities, assuming that each demand will travel to or be served by their closest remaining facility. Modeling this problem requires explicit closest assignment (ECA) constraints.

There are a number of alternative explicit constraint structures that have been proposed and used to force CA within a location model construct (Gerrard and Church 1996). Even though a number of single-level ECA constraint forms have been proposed in the literature, there exists the need for a generalized form that can address multilevel assignments. The concept of multiple levels of CA was first proposed by Weaver and Church (1985) in the development of the Vector Assignment p-median Problem (VAPMP). In their original model, they assumed that only a certain percentage, for example, 70 percent of users, will go to their first closest facility, 20 percent will go to their second closest facilities, and so on. They called this a vector of assignments. They proved that when the assignment vector percentages are nonincreasing in terms of the order of first closest, second closest, and so on, the VAPMP model will automatically allocate the specified percentage of a given demand to the first closest facility, the second closest facility, and so on, without needing to explicitly enforce CA constraints. Church and Weaver (1986) also demonstrated that the VAPMP was a very general model form that could be used to solve a wide variety of multilevel covering problems as well. The VAPMP construct has subsequently been used and extended in a number of location models involving backup service (Narasimhan, Pirkul, and Schilling 1992), reliable service (Snyder and Daskin 2005) and franchise expansion (Kolli and Evans 1999). However, similar to the original VAPMP model, these models do not explicitly use CA constraints and rely instead on the nature and structure of the specific models to ensure CA at each level. Up to this point in time, no workable constraint structure[1] has been proposed for the case when the assignment vector is not monotonically decreasing or for the general case when explicit constraints are needed to enforce CA when multiple levels are involved.

The goal of this article is to present several generalized constructs of ECA constraints in terms of multiple levels of closeness that can be used in the general form of the vector assignment median problem and related problems. These new constructs are potentially useful in modeling contexts for a wide variety of problems in which the closest facilities are not always available due to various reasons such as, busyness and capacity limits, the coexistence of spatial and nonspatial preferences, random facility failure, hostile interdiction by an enemy, and so on. Under such circumstances, certain users have to go to their second closest or even the lth closest facility for service. We also test the proposed constructs in solving a new form of the RIM model.

In the next section, we will give a brief review of past work on ECA constraints. Following that we will discuss the use of multiple-level assignment in location modeling. Next, we will show how previously proposed single-level ECA constraints can be generalized to apply to the multiple-level case. In a subsequent section, we will present a new model called the r-Interdiction Vector Assignment Location (RIVAL) problem, which requires multilevel CA constraints. This problem is a realistic extension of the RIM problem. Following that, we will present computational experience in solving the RIVAL model and compare the efficacy of each of the generalized constraint forms. Finally, we will conclude with a summary of the results and point out possible future work.

Background on CA Constraints

Location-allocation models involve the location of one or more facilities and the allocation of facility-provided services to a set of demand points. A major issue in model development is the form in which the allocation is to be accomplished. As discussed in the introduction, many models have been developed where it is assumed that each demand will be assigned to their closest facility. The classic p-median problem is an example where it is assumed that each user will be served by his or her closest facility, and this can be accomplished without explicit constraints enforcing CA. In more complicated contexts, CA constraints must be specified explicitly in the formulation. For example, if there are high variable costs associated with using specific facilities, it may be profitable from a system optimal perspective to shift the assignment of demand from a nearby highly used facility location to less frequently used sites to lower total variable cost. In fact, the earliest proposed CA constraint was developed by Rojeski and ReVelle (1970) when they expanded the p-median problem by considering a more general resource limit that includes fixed and variable costs for opening and using facilities. They argued that one consequence of expressing the more general problem is that customers are no longer automatically assigned to their closest facilities without explicit constraints. To describe the Rojeski and ReVelle construct, consider the following notation:

$i =$ an index to represent demand points, where $i = 1, 2, 3, .., n$ and $i \in I$

$j =$ an index to represent potential facility sites where $j = 1, 2, 3, \ldots, m$ and $j \in J$

$d_{ij} =$ the shortest distance between i and j

$C_{ij} = \{q \in J | d_{iq} < \mathrm{d}_{ij}$ or $\mathrm{d}_{iq} = \mathrm{d}_{ij}$ when $q < j\}$, the set of facility sites that are closer to demand i than facility j or are equidistant to i with facility j but having a lower site index than j

$$x_{ij} = \begin{cases} 1, \text{if demand at } i \text{ assigns to a facility at } j \\ 0, \text{otherwise} \end{cases}$$

$$y_j = \begin{cases} 1, \text{if a facility is located at site } j \\ 0, \text{otherwise} \end{cases}$$

Virtually all single-level models (e.g., the classic p-median problem) maintain that each demand must assign to a facility. This is established in the following assignment constraint:

$$\sum_{j \in J} x_{ij} = 1, \text{for each } i \in I \quad (1)$$

This constraint forces each demand to assign exactly once, to some located facility. In the remainder of this section, we will assume that this constraint is required in the problem being solved. From the above notation, we can formulate the Rojeski–ReVelle (1970) ECA constraint as follows:

$$x_{ij} \geq y_j - \sum_{q \in C_{ij}} y_q \text{ for each } i \in I \text{ and each } j \in J \quad (2)$$

Constraint (2) ensures that a given demand i will assign to the closest located facility to i. Suppose that the closest facility to i is located at site j, which means that $y_j = 1$ and $\sum_{q \in C_{ij}} y_q = 0$. For this case, constraint (2) equals to $x_{ij} \geq 1$, which means that i must assign to j. If one or more facilities are located closer than j, then $\sum_{q \in C_{ij}} y_q \geq 1$ and then the left-hand side of constraint (2) will be zero or negative, allowing $x_{ij} = 0$. Thus, demand i is forced by constraint (2) to assign to its closest located facility. There is one nuance that is important to discuss that was originally raised in Gerrard and Church (1996). This pertains to the case when a site q exists that is equidistant to demand i as facility site j (i.e., $d_{ij} = d_{iq}$). When this case occurs, we can arbitrarily define the site with the lowest index to be closer, so that any ties in closest distance are broken. Without this condition being added to the definition of set C_{ij}, then set $C_{ij} = C_{iq}$ and this means that sites j and q cannot be simultaneously selected for facilities as constraint (2) will force assignment to both facilities, which violates condition (1).

Wagner and Falkson (1975) proposed several location models aimed at maximizing social welfare or consumer surplus. In one of their models for a "serve-all-comer" setting, they stated that system economic efficiency may lead to the assignment of users to non-closest facilities and developed a second distinct form of CA constraints to prohibit non-closest assignment. This is structured as follows:

$$y_j + \sum_{q \notin \bar{C}_{ij}} x_{iq} \leq 1 \text{ for each } i \in I \text{ and each } j \in J \quad (3)$$

where $\bar{C}_{ij} = \{q \in J | d_{iq} < \mathrm{d}_{ij} \text{ or } \mathrm{d}_{iq} = \mathrm{d}_{ij} \text{ and } q \leq j\}$. $\bar{C}_{ij}$ represents the set of sites that are strictly closer to i than site j, plus site j itself. That is, $\bar{C}_{ij} = C_{ij} + \{j\}$. This constraint forces all assignments to more distant facilities to equal zero when a facility has been located at j (i.e., $y_j = 1$). Thus, an assignment must be to a site that is as close or closer than j when $y_j = 1$. Whereas, the Rojeski–ReVelle (R–R) constraint

(2) ensures that demand assignment will be to the closest located facility, the Wagner–Falkson (W–F) constraint will not without adding constraints that limit assignment to only sites that have been selected for facility placement:

$$x_{ij} \leq y_j \text{ for each } i \in I \text{ and each } j \in J \tag{4}$$

This type of constraint is called a Balinski (1965) constraint. Thus, when solving a model requiring CA constraints, the form proposed by Rojeski and ReVelle requires considerably fewer constraints than that proposed by Wagner and Falkson.

A third form of ECA constraint was formulated by Church and Cohon (1976) when they developed a set of models for locating regional energy facilities. In one of their formulations, they proposed that perceived safety associated with a nuclear power plant was a function of the distance to the closest located plant. Maximizing safety was then based on maximizing the weighted distances that population centers were from their closest located nuclear power plant. Without ECA constraints, an optimization model for this problem setting will assign each demand center to their farthest nuclear facility. However, it is the closest facility that actually creates the highest negative impact on an inhabitant, and therefore the assignment should be made to the closest noxious facility. They did this by imposing the following type of constraint:

$$\sum_{q \in \bar{C}_{ij}} x_{iq} \geq y_j \text{ for each } i \in I \text{ and each } j \in J \tag{5}$$

If a given site has been selected for a facility, $y_j = 1$, then this constraint forces an assignment for demand i to be made that is either to the facility at j or to a closer facility. Like the Wagner and Falkson ECA constraint, it is necessary to include a form of the Balinksi constraints (4) for this constraint to work appropriately. If assignments can be made to only those sites selected for facilities, then the above type of constraint will force the assignment to be to the closest located facility for a given demand. It is interesting to note that Hanjoul and Peeters (1987) proposed the same form of ECA constraints as Church and Cohon for an expanded version of the Simple Plant Location Problem (SPLP) that incorporated user's preferences in assignment, which may or may not be a function of distance or cost. Another example of a semi-obnoxious facility is that of a fire station, where it is beneficial to be close but not too close to avoid the noise of sirens, and so on (see Church and Roberts 1983; Murray et al. 1998). Church and Roberts used the same form of ECA constraints to ensure that impacts and benefits were appropriately accounted for in each neighborhood.

In an entirely different setting, Dobson and Karmarkar (1987) investigated the problem of locating facilities in a competitive environment where a company has enough resources to locate facilities so that it would be unprofitable for any entering competing company to establish new facilities. They assumed that customers will

Table 1. Summary of Existing Explicit Closest Assignment Constraints

Abbreviation	Reference	Equation # in Text	Balinski Type Constraint Required
R–R	Rojeski and ReVelle(1970)	2	No
W–F	Wagner and Falkson(1975)	3	Yes
C–C	Church and Cohon (1976)	5	Yes
D–K	Dobson and Karmarkar (1987)	6	Yes

choose to go to their closest facility. In their model, Dobson and Karmarkar proposed a fourth type of ECA constraints:

$$x_{ij} + y_q \leq 1 \text{ for each } i \in I, j \in J \text{ and } q \in C_{ij} \tag{6}$$

This constraint specifies that demand i cannot assign to a facility at j if a closer site, q has been selected for a facility. This type of constraint does not prevent an assignment to j if no closer facilities have been located, even when site j has not been selected. Thus, like the W–F and Church–Cohon constructs, this form also requires the use of the Balinski constraints (4). It is also important to state that the number of Dobson–Karmarkar (D–K) constraints is considerably larger than the numbers of other types of ECA constraints proposed in the literature. Overall, there are four basic forms of ECA constraints for discrete assignment that have been proposed in the literature to force CA. Table 1 summarizes each of the constraint forms, which will be referred to by abbreviation in subsequent sections.

Multilevel CA

It is possible during system operation that a given facility may not be available due to a variety of circumstances (Daskin 1983). This possibility was first explored within the p-median problem structure by Weaver and Church (1985) in a problem dealing with ambulance location. Suppose that ambulances are placed at dispatching posts across a city. When a call for service is generated, it is common to dispatch the closest available vehicle as some may be busy handling prior service calls. For example, suppose that a given demand might be served 70 percent of the time by its closest ambulance dispatch post, 20 percent of the time by its second closest dispatch post, and 10 percent of the time by its third closest dispatch post, then it is necessary to track assignments to not only the closest dispatch post but the second closest post and the third closest post. To handle such a case, Weaver and Church (1985) formulated the VAPMP, where each demand i has an associated assignment vector, for example, [0.7, 0.2, 0.1], which represents the fraction of the time that a demand is served by its first closest facility, second closest facility, and so on. These fractions (or percentages) may reflect the unavailability of servers or user preferences and they can be determined from empirical data, by simulation or analytically (Weaver and Church 1985). The VAPMP problem is a generalized form of the p-median problem, where the assumption of CA has been relaxed, and is now represented by multiple levels of assignment in terms of closest, second closest, and so on.

The construct of multilevel assignment has since been investigated in a number of applications. Pirkul and Schilling (1989) developed a capacitated maximal covering model with backup and used two sets of assignment variables, one for primary coverage and one for backup coverage. Narasimhan, Pirkul, and Schilling (1992) extended the work of Pirkul and Schilling by adding the capability of handling multiple levels of backup. For example, if the primary facility is unavailable, the user is then served by the first backup facility, and if the first backup is unavailable, the user is served by the second backup and so forth. From a different perspective, Snyder and Daskin (2005) proposed a location model where some facilities may be unreliable and not always available and other facilities are always reliable. Suppose for the sake of explanation that all facilities are unreliable and can fail and be unavailable for service with a failure probability of q. This would mean that a demand can be served by its closest facility $(1 - q)$ fraction of the time. The probability that the $l - 1$ -closest facilities fail can be computed as q^{l-1}. Thus, the chance that a demand will be served by his or her lth closest unreliable facility is $q^{l-1}(1 - q)$. Snyder and Daskin incorporated the probabilities of being unavailable into a p-median framework using multiple levels of assignment. The objective was to minimize expected travel distances or costs.

While locating franchise branches, Kolli and Evans (1999) argued that the fraction of demand from a neighborhood that shops at a specific store depends on geographic proximity as well as non-geographic factors such as price, economic status, and travel patterns (e.g., trip chaining, shop on the way home from work, etc.). They made an assumption that only a certain percentage of customers or all customers a certain percentage of the time will go to their closest store, at other times they will travel to a second closest store, and so on. In fact, this is a basic tenet in the gravity model that has been used for customer assignment in retail location (Huff 1964).

Even though multilevel assignment has been used in a number of location model constructs, a question left open is how to efficiently enforce these constraints/conditions. The models that have appeared in the literature so far rely on certain properties of the problem such as the sense of the objective function and the value of assignment weights, so that explicit constraints are not necessary. For example, Snyder and Daskin (2005) did not use constraints to enforce multilevel CA when unreliable facilities fail, as second closest, third CA, and so on, will occur in the appropriate order as long as the weights or probabilities associated with farther assignments decreases as the order of closeness increases. This same issue is true for the VAPMP of Weaver and Church (1985) when the assignment vector values decrease with increasing closest order (1st, 2nd, 3rd, etc.). Weaver and Church did acknowledge that when the vector was not monotonically decreasing with increasing order, explicit constraints are necessary to enforce appropriate order. Although past work involving multilevel assignment has focused on cases where ECA constraints are not necessary, a number of problem settings do exist where explicit constraints are necessary (e.g., obnoxious, retail, and interdiction), regardless of the probabilities of assignment. For example, consider the simple RIM problem where the objective is to maximize the total travel distance by interdicting or removing r facilities. In such a context, the objective function no longer encourages CA without explicit constraints (Church, Scaparra, and Middleton 2004). If this problem were cast within a multilevel assignment context, then constraints would be necessary to enforce CA for multiple levels of closeness as well. In the next section, we will show how structures that have been developed to enforce single-level CA can be generalized to enforce multilevel CA.

Formulating Multilevel CA Constraints

Past research on multilevel assignment has focused on cases where ECA constraints have not been necessary. For all practical purposes, no workable multilevel ECA constraints have been proposed. We show here that the four single-level forms can each be generalized to one or several multilevel forms. Testing of these generalized forms will be given in a subsequent section. To start, we need to expand our notation somewhat to address multiple assignment cases:

l = an index that represents the level of CA where $l = 1, 2, 3, \ldots L$

L = the maximum number of levels being considered in the model

$$x_{ij}^{l} = \begin{cases} 1, \text{if demand } i \text{ assigns to facility } j \text{ as the } l\text{th closest open facility} \\ 0, \text{otherwise} \end{cases}$$

Consider the basic case where each demand must assign to a set of facilities in order of closeness. Regardless of the exact model, we need to require a discrete assignment be made to a facility for each level of closeness. This can be defined as follows:

$$\sum_{j \in J} x_{ij}^{l} = 1 \text{ for each } i \in I \text{ and each } l = 1, 2, 3, \ldots, L \quad (7)$$

This type of constraint can be viewed as a general form of constraint (1). We also need to ensure that a given demand assigns at most once to a specific facility across all orders. This can be accomplished with the following set of conditions:

$$\sum_{l=1}^{L} x_{ij}^{l} \le y_j \text{ for each } i \in I \text{ and each } j \in J \quad (8)$$

This constraint is a generalized form of the Balinski constraint (5) and ensures that a given demand assigns to a facility at most once in terms of closeness order (Weaver and Church 1985). Except for a few cases as noted below, we will assume that these two constraints are used to require assignment at each level, prevent an assignment to only sites chosen for facilities, and prevent a facility from serving a demand for more than one assignment order.

Generalizing the Rojeski and ReVelle Constraint Form

We can generalize the R–R constraint given in equation (2) into the following form:

$$x_{ij}^l \geq y_j - \sum_{q \in C_{ij}} y_q + \sum_{s<l} \sum_{q \in C_{ij}} x_{iq}^s - \sum_{s<l} x_{ij}^s, \text{ for each } i \in I, j \in J, \quad (9)$$

$$\text{and } l = 1, 2, 3, \ldots, L$$

Constraint (9) forces x_{ij}^l to be 1, that is, j to be the lth closest facility with respect to demand i, when the following factors are true:

i. a facility is open at site j, that is, $y_j = 1$
ii. all open facilities that are closer than facility j (where the sum $\sum_{q \in C_{ij}} y_q$ represents how many facilities are located closer than j) are matched up exactly by assignments to these closer facilities for assignment levels s where $s < l$ ($\sum_{s<l} \sum_{q \in C_{ij}} x_{iq}^s$), and
iii. demand i has not already been assigned to site j as something closer than the lth closest open facility (this means that $\sum_{s<l} x_{ij}^s = 0$).

It is important to note that the generalized Balinski constraint (8) will prevent any assignments to site j unless a facility has been located at that site (i.e., $y_j = 1$). Constraint (9) itself acts in the same manner for the first closest facility assignment as the original R–R constraint (2). Consider the case where the first CA has been made. The constraint forces assignment to the next closest facility as the second closest, when $l = 2$ as the right-hand side of the condition will equal 1, which represents the net of: $1 - 1 + 1 - 0 = 1$. This net sum represents the sum of the location selection variables *minus* the number of locations selected that are located closer than this site *plus* the number of closer assignments that have been made (up to but not including level l) *minus* the number of assignments already made to site j. This net sum will never exceed 1 and will equal 1 only when site j is the 2nd closest (and in general the lth closest) facility to demand i. Note[2], that it is required to have one constraint for each i, j pair and each level l. We will refer to this generalized form as R–R1[3].

It is also possible to generalize the R–R constraint form in a more compact manner as follows:

$$\sum_{l=1}^{L} (L + 1 - l) x_{ij}^l \geq L \cdot y_j - \sum_{q \in C_{ij}} y_q, \text{ for each } i \in I \text{ and } j \in J \quad (10)$$

Note that this constraint (called R–R2) works with one constraint for all assignment levels for a given i,j pair. Consider the case when $L = 3$ for a given demand i and facility site j:

$$3x_{ij}^1 + 2x_{ij}^2 + x_{ij}^3 \geq 3y_j - \sum_{q \in C_{ij}} y_q, \text{ for each } i \in I \text{ and } j \in J \tag{11}$$

It is easy to see the above constraint is only effective when $y_j = 1$, otherwise the left-hand side would at most be zero in value. Let $Q = \sum_{q \in C_{ij}} y_q$, the number of open facilities that are closer than j. Recall that by constraint (8), at most one of the x_{ij}^l can be 1 over all values of l. When $Q = 0$, it can be easily verified that only x_{ij}^1 can be 1. When $Q = 1$, some facility k closer than j must have been assigned as the first closest, that is, $x_{ik}^1 = 1$. By assignment constraint (7), j cannot be another first closest facility, thus $x_{ij}^1 = 0$. So, the constraint nicely reduces to (given $y_j = 1$and j is not the first closest site):

$$2x_{ij}^2 + x_{ij}^3 \geq 3y_j - 1 = 2 \tag{12}$$

So, if j is the second closest site, then the above constraint would then force $x_{ij}^2 = 1$. Similarly, when $Q = 2$, there must have been some facilities k and t, such that $x_{ik}^1 = 1$ and $x_{it}^2 = 1$. Again, only x_{ij}^3 can equal 1 as demand i has already assigned to its first and second closest sites. The effective form of the constraint reduces to $x_{ij}^3 \geq 3y_j - 2$ which forces $x_{ij}^3 = 1$ when $y_j = 1$. The nice advantage of this second form is that it requires only one constraint for each i,j pair, which is L times smaller than the form of R–R1.

There is a third alternative in which to generalize the R–R constraints for a multi-level CA problem. This can be accomplished as a cascading set of constraints similar to R–R2 (10) for all values of l from 1 to L:

$$\sum_{s=1}^{l} (l + 1 - s) \cdot x_{ij}^s \geq l \cdot y_j - \sum_{q \in C_{ij}} y_q, \text{ for each } i \in I, j \in J \text{ and } l = 1, 2, 3, \ldots, L \tag{13}$$

This will be designated as the R–R3 form. As an example to help demonstrate how such constraints work, consider the above constraint form written for cases $l = 1$, 2, and 3 as follows:

$$x_{ij}^1 \geq y_j - \sum_{q \in C_{ij}} y_q, \text{ for each } i \in I \text{ and each } j \in J \tag{13a}$$

$$2x_{ij}^1 + x_{ij}^2 \geq 2y_j - \sum_{q \in C_{ij}} y_q, \text{ for each } i \in I \text{ and } j \in J \tag{13b}$$

$$3x_{ij}^1 + 2x_{ij}^2 + x_{ij}^3 \geq 3y_j - \sum_{q \in C_{ij}} y_q, \text{ for each } i \in I \text{ and } j \in J \tag{13c}$$

Consider for any given set of open facilities $F = \{j|y_j = 1, j \in J\}$, the first constraint (13a) will ensure that the first closest facility j will force $x_{ij}^1 \geq 1$. If j is the closest site and $x_{ij}^1 = 1$, then all higher ordered conditions (13b) and (13c) will automatically be met for site j. This is due to the fact that the coefficient for the variable x_{ij}^1 is large enough to satisfy conditions (13b) and (13c) when $x_{ij}^1 = 1$. By constraint (7), all other facilities k that are not the closest to i will have a first CA of $x_{ik}^1 = 0$. Let us suppose that site j is the second closest facility to demand i. Then $y_j = 1$ and $\sum_{q \in C_{ij}} y_q = 1$. Note that constraint (13a) will have a right-hand side value of 0, so there is no requirement on first CA. Also note, the assignment constraint (7) restricts the number of first CAs to be only one, so x_{ij}^1 must be zero. Thus, the remaining conditions boil down to:

$$x_{ij}^2 \geq 2y_j - \sum_{q \in C_{ij}} y_q \qquad (13b')$$

$$2x_{ij}^2 + x_{ij}^3 \geq 3y_j - \sum_{q \in C_{ij}} y_q \qquad (13c')$$

Then constraint (13b′) boils down to $x_{ij}^2 \geq 1$ as the right-hand side will equal $2 - 1 = 1$. The minus 1 is associated with the subtraction of the sum of facilities that have been sited which are closer than j is to demand i. Equation (13c′) will be met when $x_{ij}^2 = 1$. Thus, if site j is the second closest, the constraints will force a second CA to j. The logic is similar in showing that if site j is the third closest site; then condition () will force as (13a) and (13b) will be met by the previously constrained first and second CAs. Thus, the "cascading set" of constraints will force a demand to assign to facilities in order of first closest, second closest, third closest, and so on. Obviously, this cascading set (13) has L times more ECA constraints than constraint (10). But the reward is that this reduction does not rely on the existence of the generalized Balinski constraint (8).

As discussed in Gerrard and Church (1996), when the original R–R ECA constraint (2) is used with assignment constraint (1), Balinski constraints (4) are redundant. In a similar fashion, it is easy to show that if we use the cascaded form (13), constraint (8) becomes redundant. It can also be verified that the R–R1 constraint set (9) can also be combined with the assignment constraint (7) to force $x_{ij}^l \geq 1$ without needing constraint (8) to prevent an assignment to a location that has not been selected for a facility.

It is well known that Balinski constraints are integer friendly. While adding them increases the size of the model, they sometimes improve the integer friendliness for the whole model when using a general purpose integer programming software package. Another property of ECA constraints discussed by Gerrard and Church (1996) is whether the constraint automatically forces the assignment variable to be 0, 1 integer. It is not difficult to verify that constraints (9), (10), and (13) force $x^l_{ij} \geq 1$ for the appropriate facilities and that the generalized assignment constraint (7) forces other x^l_{ik} to be 0. Therefore, the generalized constraints (9), (10), and (13) also enforce full assignment.

Generalizing the Wagner and Falkson Constraint Form

The original W–F ECA constraint (3) basically prevented assignments farther than a sited facility to a given demand. Since the constraint associated with the closest facility to a given demand prevented assignments to any site that was farther than this closest facility, and since Balinski constraints (4) were included, assignment would be forced to the closest facility. This type of property can be generalized as follows:

$$\sum_{q \in \overline{C}_{ij}} y_q + M \cdot \sum_{q \notin \overline{C}_{ij}} x^l_{iq} \leq M + l - 1 \text{ for each } i \in I, j \in J, \text{ and for } l = 1, 2, 3, \ldots, L \quad (14)$$

where M is a sufficiently large number (e.g., $M = |\overline{C}_{ij}|$). Similar in principle to original W–F constraint, this generalized constraint (14) states that if l or more facilities are already located closer than or equidistant with j to demand i, any site beyond j cannot be assigned as the lth closest facility. Here the generalized Balinski constraints (8) are needed to ensure that an assignment is made to only those sites selected for facilities. What is important to realize is that for each value of l, the most restrictive constraint for demand i is associated with the lth closest facility to that demand. Such a restriction, in concert with constraints (7) and (8) force the lth CA to the lth closest facility. One should recognize that these constraints may not be very "integer friendly," since the value of M may tend to yield fractional assignments in the relaxed problem, which might require that branch and bound be used to a greater extent in solving a problem with IP software.

Generalizing the Church and Cohon (C–C) Constraint Form

The C–C constraint (5) is based on the premise that either an assignment of demand i must be made to a facility sited at j or that the assignment is made to a facility that is even closer. The most constraining factor for a given demand will occur associated with the closest facility to that demand. Again, Balinski constraints are needed to ensure that assignments are made only to those sites selected for facilities. One can think of the W–F and C–C constraints as being opposites: one forces assignment of a demand to a site that is as close or closer to a given facility and the other prevents

Table 2. Properties of the Generalized Multilevel Explicit Closest Assignment (ECA) Constraints

Generalization	# of Constraints	$\sum_{l=1}^{L} x_{ij}^l \le y_j$ Necessary	x_{ij}^l Always 0,1
R–R1 (9)	$L \cdot n^2$	No	Yes
R–R2 (10) (non-cascaded)	n^2	Yes	Yes
R–R3 (13) (cascaded)	$L \cdot n^2$	No	Yes
W–F (14)	$L \cdot n^2$	Yes	No
C–C (15)	$L \cdot n^2$	Yes	Yes
D–K (16)	$L \cdot n^2$	Yes	No

assignments that are farther away than a given located facility. We can generalize the Church-Cohon ECA constraints as follows:

$$\sum_{q \in \overline{C}_{ij}} x_{iq}^l + (x_{ij}^{l-1} + \ldots + x_{ij}^1) \ge y_j \text{for each } i \in I,\ j \in J \text{ and each } l = 1, 2, 3, \ldots, L \quad (15)$$

This constraint ensures that if a facility is located at j and j is not the $1\text{st}, 2\text{nd}, \ldots, (l-1)^{st}$ closest facility, then either j or some site closer than j is the lth closest. Observe that Balinski constraints (8) are needed to prevent the assignment of demand i to sites in $\overline{C}_{ij}$ that have not been selected for facility placement.

Generalizing the Dobson and Karmarkar Constraint Form

The original Dobson and Karmarkar (1987) constraints (6) were based on a "pairwise" property: either an assignment could be made for demand i to a facility at site j or a facility had been located at site q, which was closer to i than j. The D–K constraints can be generalized as follows:

$$\sum_{q \in C_{ij}} y_q + M \cdot x_{ij}^l \le M + l - 1 \text{ for each } i \in I,\ j \in J \text{ and each } l = 1, 2, 3, \ldots, L \quad (16)$$

where M is a sufficiently large number (e.g., $M = |C_{ij}|$). Constraint (16) sates that if l facilities have been located between demand i and site j, then demand i cannot assign to site j as the lth closest facility assignment.

Gerrard and Church (1996) presented a detailed review of single-level CA constraints and the properties of each ECA constraint form. From the previous discussion, we can see that many of the properties of single-level ECA constraints discussed in Gerrard and Church extend to multilevel forms. For the ease of comparison, detailed results similar to those in Gerrard and Church (1996) are summarized below in Table 2.

The *r*-Interdiction Vector Assignment Location (RIVAL) Problem

In this section, we present an application that requires multilevel ECA constraints. This application allows us to test the efficacy of the generalized multilevel ECA constraints that were developed in the last section. Although there are several possible problems in which we can test the multilevel ECA constraints, including the general form of VAPMP, we have chosen as a candidate a general form of the RIM problem (see Church, Scaparra, and Middleton 2004; Church and Scaparra 2007; Scaparra and Church 2008). The RIM problem is defined as follows:

> Identify the set of r facilities that, if removed from a system of p facilities, results in the highest level of weighted distance, assuming that each demand is served by their closest remaining facility.

The major difference between an RIM problem and a p-median problem is that the p-median problem involves designing an efficient system whereas the interdiction problem involves maximally disrupting a system (in short minimize weighted distance by locating p facilities vs. maximize weighted distance by removing r facilities). For the RIM problem, ECA constraints are necessary. If ECA constraints are not used a model will assign each user to the farthest facility possible rather than the closest open facility after interdiction. Suppose that on the average, a demand is served by its closest facility 80 percent of the time and a second closest facility 20 percent of the time or some other set of percentages to closest facility, second closest, and so on. To address this within the context of interdiction, we can extend the RIM problem definition to include multilevel assignments as follows:

> Identify the set of r facilities that, if removed from a system of p facilities, results in the highest level of weighted distance, assuming that each demand is served by their closest remaining facility a certain percentage of the time, by its second closest remaining facility a certain percentage of the time, and so on.

We will call this the RIVAL problem. To formulate the RIVAL problem, we need to introduce some additional notation:

a_i = a measure of demand at i
b_{il} = the fraction of time that demand at i will travel to or be served by their lth closest Facility
d_{ij} = the distance between demand i and facility j.
F = the set of existing facilities.

$$\bar{y}_j = \begin{cases} 1, & \text{if site } j \text{ remains after interdiction} \\ 0, & \text{otherwise} \end{cases}$$

We can then define the RIVAL model as follows:

$$\text{Max } Z = \sum_{l=1}^{L} \sum_{i \in I} \sum_{j \in F} a_i \, d_{ij} \, b_{il} \, x_{ij}^{l} \tag{17}$$

Subject to:

$$\sum_{j \in F} x_{ij}^{l} = 1 \quad \text{for each } i \in I \text{ and each } l = 1, 2, 3, \ldots, L \tag{18}$$

$$\sum_{j \in F} \bar{y}_j = p - r \tag{19}$$

plus constraints that ensure multilevel CA.

The objective function maximizes the total weighted distance after interdiction assuming that each demand will be served a certain fraction of the time by their closest remaining facility, a certain fraction of the time by their second closest remaining facility, and so on. If r facilities are interdicted, then $p - r$ facilities remain after interdiction. Here the problem is posed in terms of which $p - r$ facilities should be left open instead of which r facilities are to be removed to disrupt weighted distance the most. The first constraint (18) is the assignment constraint, which ensures that each demand assigns to a facility for each level of closeness. The second constraint (19) states that only $p - r$ facilities will remain after interdiction. Finally, it is necessary to append to this formulation multilevel ECA constraints. We can potentially use any of the multilevel ECA constraints presented in Table 2 (including generalized Balinski constraints (8) when needed). Since the $\bar{y}_j$ is essentially a location selection variable (identifying those sites kept open), all of the constraints in the previous section can be employed using the $\bar{y}_j$ variable instead of a y_j variable without any loss of generality.

One caveat in using the generalized Balinski constraint (8) is worth attention. In general, constraint (8) is required for most multilevel assignment location problems to prevent a demand from being assigned to an unopened facility. For example, if the Balinski constraint is removed in solving the vector assignment p-median location problem, the generalized C–C constraint (15) may well allow demand i to assign to an unopened facility that is closer than j. In fact, the objective of the vector assignment median problem will encourage the closest possible assignments, even to sites not selected for facilities unless constraints (8) are used. However, in the RIVAL problem, the objective function encourages assignments to the farthest possible facilities. Therefore, assignments to closer but unopened facilities are discouraged and will not occur in an optimal solution to a RIVAL model. In addition, more than one multilevel assignment to a given facility will not occur, even without the Balinski constraint (8), because total weighted distance will be higher by not doing so for the RIVAL problem. Therefore, Balinski constraints (8) can be omitted and the model will still be able to force the demand to assign to the correct facility. It should be noted that when distance ties are present in the data set for a given demand, the model, without Balinski constraints, may erroneously assign a demand to an unopened but equidistant facility. In this case, the model will compute the correct objective function value but may assign to unopened facilities. Unless otherwise stated, model results in the next section will be based on using both ECA and generalized Balinski constraints.

Computational Experiments

In this section, we will investigate the efficacy of the six generalized multilevel ECA constraint forms in solving the RIVAL problem. We believe that these experiments will not only demonstrate which type of multilevel ECA constraint performs best in solving the RIVAL problem, but that it will help identify which form of the multilevel ECA constraint is likely to outperform the others when used on other problems that require multilevel ECA constraints.

We tested RIVAL under a variety of parameter settings. The model was implemented in ILOG OPL 5.5 and CPLEX11, and run on an Intel 2.0GHz Xeon CPU with 5Gbytes of memory. All models were implemented according to (17)–(19) plus a specific generalized ECA constraint. In addition, the assignment variables, x_{ij}^{l}, are constrained to be integer (i.e., zero-one) only when the ECA constraints do not force full assignment (as noted in the last column of Table 2). We used two data sets in our experiments. The first is a forty-nine city data set comprised of the biggest city (based on population) in each state of the contiguous forty-eight states, plus Washington, DC (Figure 2 and Appendix). The demand weight of each city is given as the population according to U.S. Census. Intercity Euclidean distances were estimated in km. on a contiguous U.S. conic-equidistant projection (using ESRI ArcGIS 9.3 software). The second data set is comprised of 150 U.S. cities taken from a data set used by Scaparra, Liberatore, and Daskin (2008).

We solved the RIVAL model on both data sets for a range of r varying from 1 to 7, and with two different assignment vectors of [0.7, 0.2, 0.1] and [0.6, 0.4]. The existing facilities for both problem sets ranged from 10 to 25 and were selected by solving a vector assignment median model with the same assignment vector. Most of the results given here are confined to the $p = 15$ case to keep the number of tables to a manageable level and size and since the computational trends associated with the $p = 15$ results are representative of the other tests. For the vector assignment median problem, each city served as both a demand and a potential facility site. Each RIVAL problem associated with a specific assignment vector and r value was solved six times, one for each of the ECA generalizations. The running statistics for the RIVAL model tests on the forty-nine city data set are given in Tables 3 and 4. Table 3 presents the RIVAL problem using the assignment vector

Table 3. Computational Times in Solving RIVAL on the Forty-Nine City Data Set with Assignment Vector [0.7, 0.2, 0.1] and $p = 15$

r	Objective	R–R1	R–R2	R–R3	W–F	D–K	C–C
1	7.6681436e9	0.25	0.06	0.19	0.25	0.06	0.56
2	9.539833e9	1.11	1.03	1.58	20.55	14.56	1.81
3	1.555025e10	1.28	1.22	2.18	11.76	16.08	1.61
4	1.8379373e10	2.07	1.51	3.65	19.94	62.77	1.61
5	2.10066e10	3.92	2.82	5.77	29.13	78.27	2.53
6	2.55659e10	4.04	3.68	5.79	126.75	83.91	4.07
7	3.1855745e10	3.18	2.70	5.10	67.67	33.10	3.07
Ave.	n/a	2.26	1.86	3.46	39.43	41.25	2.18

Note: The column for each form of explicit closest assignment (ECA) gives the computation times in seconds. The average time for a given constraint structure is given in the row labeled *Ave.*

Table 4. Computational Times in Solving RIVAL on the Forty-Nine City Data Set with Assignment Vector [0.6, 0.4] and $p = 15$

r	Objective	R–R1	R–R2	R–R3	W–F	D–K	C–C
1	9.392304e9	0.11	0.05	0.11	0.14	0.05	0.16
2	1.5861786e10	0.34	0.25	0.39	1.28	1.75	0.62
3	1.9176632e10	0.45	0.40	0.50	6.79	5.21	0.69
4	2.0995402e10	0.75	0.76	0.95	6.68	10.31	1.04
5	2.5832415e10	0.89	0.77	1.12	11.25	12.00	1.03
6	3.1648756e10	0.92	0.87	1.17	10.14	21.45	1.03
7	3.5090354e10	1.14	1.19	1.40	12.20	17.16	1.22
Ave.	n/a	0.66	0.616	0.806	6.926	9.706	0.82

Note: The column for each form of explicit closest assignment (ECA) gives the computation times in seconds. The average time for a given constraint structure is given in the row labeled *Ave.*

of [0.7, 0.2, 0.1] and Table 4 presents computational results for the assignment vector of [0.6, 0.4]

From the above tables, we can see with few exceptions that the R–R2 generalization is faster than other ECA formulations and the R–R1, R–R3, and C–C multilevel generalizations are somewhat close to the R–R2 in computation time. The other two generalized forms, W–F and D-K, are notably more difficult to solve, especially for r values exceeding 4 where computation times were between 8 and 20 times longer than the R–R2 model form. In addition, note that the W–F and D-K constraint forms require integer restrictions on the x_{ij}^{l} variable when solving RIVAL and these additional conditions may be responsible for the increase in running times. Comparing across Tables 3 and 4, one can see that solving with an assignment vector of [0.7, 0.2, 0.1] takes more time than solving problems with an assignment vector of [0.6, 0.4]. This is quite likely due to the fact that the three component problem has

Table 5. Computational Times in Solving RIVAL on the 150 City Data Set with Assignment Vector [0.7, 0.2, 0.1] and $p = 15$

r	Objective	R–R1	R–R2	R–R3	W–F	D–K	C–C
1	1.1585537e10	1.81	0.39	1.53	1.73	0.44	4.31
2	1.4164068e10	8.21	7.78	14.52	215.80	195.72	17.68
3	1.861333e10	12.18	12.34	24.49	223.78	156.84	17.69
4	2.5777545e10	14.38	24.17	39.84	164.00	189.57	25.57
5	3.1364794e10	26.19	23.07	51.04	126.80	359.07	28.56
6	3.4557084e10	32.32	69.59	113.40	1821.70	4029.27	24.57
7	3.9903592e10	81.78	180.81	309.52	860.94	2661.41	47.55
Ave.	n/a	25.26	45.45	79.19	487.82	1084.61	23.704

Note: The column for each form of explicit closest assignment (ECA) gives the computation time in seconds. The average time for a given constraint structure is given in the row labeled *Ave.*

Table 6. Computational Times in Solving RIVAL on the 150 City Data Set with Assignment Vector [0.6, 0.4] and $p = 15$

r	Objective	R–R1	R–R2	R–R3	W–F	D–K	C–C
1	1.2970882e10	0.81	0.27	0.72	0.91	0.25	1.40
2	1.8062303e10	2.67	2.43	3.73	27.08	20.75	5.54
3	2.0889012e10	4.43	4.55	6.41	39.23	51.54	4.79
4	2.4546777e10	6.66	7.93	11.62	146.78	88.39	10.19
5	3.0709574e10	8.38	10.94	17.16	67.16	217.67	10.64
6	3.7794513e10	11.29	12.64	21.68	115.07	205.31	8.97
7	4.1446834e10	41.78	35.07	47.46	105.43	550.76	11.76
Ave.	n/a	10.86	10.55	15.54	71.66	162.09	7.61

Note: The column for each form of explicit closest assignment (ECA) gives the computational times in seconds. The average time for a given constraint structure is given in the row labeled *Ave.*

an addition level as compared to the two component one, as more levels require more variables and constraints in general (except for R–R2).

To see how the formulations scale up in terms of problem size, we also solved the RIVAL model on the larger 150 city data set. We used the same two assignment vectors. The running times are shown in Tables 5 and 6 for the fifteen facility problem.

From Tables 5 and 6, we can observe that the generalized ECA constraints of C–C, R–R1, R–R2, and R–R3 tend to outperform the generalized D–K and W–F ECA constraints for the larger problem as well. Within the former group, the generalized C–C constraints appear to yield faster times on the average than the generalized R–R constraints. Up to this point, we have presented results for only fifteen facilities. As another example in demonstrating the efficacy of each constraint form, Table 7 presents results for the $p = 25$ case on the 150 node problem

Table 7. Computational Times in Solving RIVAL on the 150 City Data Set with Assignment Vector [0.7, 0.2, 0.1] and $p = 25$

r	Objective	R–R1	R–R2	R–R3	W–F	D–K	C–C
1	7.5659433e9	4.91	0.87	3.95	2.25	0.22	31.67
2	8.549814e9	32.34	24.35	54.04	1115.13	311.88	63.04
3	1.0970256e10	49.55	78.84	166.05	1536.30	2,443.18	86.75
4	1.3565471e10	137.91	115.18	240.13	798.19	6,172.30	116.50
5	1.4777703e10	254.45	256.28	539.40	3,830.28	10,800.1*	115.02
6	1.7277186e10	470.56	406.18	813.17	2617.81	10,800.2*	140.25
7	2.0427323e10	581.29	445.88	1142.32	3758.50	10,800.1*	245.66
Ave.	n/a	218.72	189.65	422.72	1951.21	5,904.01	114.13

Note. The column for each form of explicit closest assignment (ECA) gives the computational times in seconds. The existing facility configuration is [1, 2, 3, 4, 5, 6, 7, 8, 12, 14, 16, 23, 26, 32, 35, 40, 52, 55, 56, 70, 83, 98, 115, 129, and 145]. The cells with * indicates that the optimal solution had not been found within the time limit; the average time for a given constraint structure is given in the row labeled *Ave.*

using the assignment vector [0.7, 0.2, 0.1]. In Table 7, we can see that R–R1, R–R2, R–R3, and C–C outperform W–F and D–K. Furthermore, we can see that the C–C constraint tends to outperform R–R1, R–R2, and R–R3 for higher values of r, which is basically the same trend that can be observed in Tables 5 and 6. Overall, the C–C advantage tends to increase with increasing problem size, increasing values of r, and increasing values of p; however, R–R1, R–R2, and R–R3 are usually competitive with C–C for small values of r.

Past work has shown that Balinski constraints are integer friendly and may help tighten a location model (Morris 1978; ReVelle 1993). To investigate the effect of the trade-off between adding "integer friendly conditions" at the expense of adding redundant constraints, we solved the RIVAL model for each of the formulations without the generalized Balinski constraints (8) on the 150 city data set where the assignment vector is [0.7, 0.2, 0.1]. The results are shown in Table 8 for the fifteen facility problem.

Comparing Table 8 with Table 5, we can observe that without the generalized Balinski constraints, the running times when using the R–R1, R–R2, R–R3, and C–C constraints are actually shorter in general. However, the running time for the generalized W–F and D–K constraints are significantly longer and in many instances take longer than the 3-hour time limit we set for solving any specific problem. This suggests that the former group of generalized constraint formulations are already relatively "tight" and that the Balinski constraint does not appreciably help in speeding up the solution process while for the latter group (W–F and D–K) generalized Balinski constraints assist in significantly reducing overall computational times in solving RIVAL.

Next, we investigated the solutions on the forty-nine city data set to get a deeper understanding of the characteristics of the RIVAL model. In Tables 9 and 10, we list the main characteristics of each of the solutions generated for the forty-nine city data set. Table 9 presents the solutions for the assignment vector of [0.7, 0.2, 0.1] and

Table 8. Computational Times in Solving RIVAL on the 150 City Data Set without Balinski Constraints with Assignment Vector [0.7, 0.2, 0.1] and $p = 15$

r	Objective	R–R1	R–R2	R–R3	W–F	D–K	C–C
1	1.1585537e10	1.14	0.20	0.91	0.47	0.08	13.28
2	1.4164068e10	6.86	2.62	10.61	983.17	6,901.80	16.25
3	1.861333e10	10.61	5.38	22.73	6,581.09	4,234.01	19.20
4	2.5777545e10	12.37	21.25	17.18	2,615.82	10,800.01*	27.97
5	3.1364794e10	17.18	38.06	32.79	10,800.04*	2,759.81	20.30
6	3.4557084e10	26.63	44.54	64.71	10,800.04*	10,800.04*	24.88
7	3.9903592e10	51.31	78.87	86.35	10,800.01*	10,800.11*	25.26
Ave.	n/a	18.01	27.27	33.61	6,082.95	6,613.69	21.02

Note: The column for each form of explicit closest assignment (ECA) gives the computational times in seconds. The cells with *indicate that the optimal solution had not been found within the time limit; The average time for a given constraint structure is given in the row labeled *Ave.*

Table 9. Site Selections for the Forty-Nine City Data Set with Assignment Vector [0.7, 0.2, 0.1] and $p = 15$

r	Open Sites	Closed Sites	Objective
0*	1 2 3 7 9 14 15 20 25 30 31 36 41 42 43	None	6.1428756e9
1	2 3 7 9 14 15 20 25 30 31 36 41 42 43	1	7.6681436e9
2	2 7 9 14 15 20 25 30 31 36 41 42 43	1 3	9.539833e9
3	1 2 3 7 9 14 15 20 25 30 31 36	41 42 43	1.555025e10
4	1 2 3 7 9 14 15 20 25 30 31	36 41 42 43	1.8379373e10
5	1 2 3 7 9 14 15 20 25 31	30 36 41 42 43	2.10066e10
6	1 2 3 7 9 14 15 20 25	30 31 36 41 42 43	2.55659e10
7	1 2 3 7 9 14 15 20	25 30 31 36 41 42 43	3.1855745e10

Note: $r = 0$, represents the no interdiction case, that is, the original vector assignment median solution for the associated assignment vector of [0.7, 0.2, 0.1].

Table 10 presents the solutions for the assignment vector of [0.6, 0.4]. For each interdiction value of $r = 0,1,2,\ldots,7$, we give the objective value as well as present lists of which sites are open and which sites are closed. Note that the case of $r = 0$ represents the base set of facilities before interdiction. Table 9 indicates that the base case of weighted distance before any interdiction is approximately 6.14 billion (people-kilometers) and for the case of $r = 3$, sites 41, 42, and 43 are interdicted with a weighted distance of over 15 billion (people-kilometers).

Figure 1 presents a plot of the solutions in Tables 9 and 10 in terms of weighted distance versus the level of interdiction, r. It is interesting to note that even though weighted distance increases with each successive strike, the marginal level of increases tends to fluctuate. For example, the $r = 3$ solution for [0.7, 0.2, 0.1] has a higher average weighted distance per interdicted facility than the $r = 2$ solution (notice the abrupt change in slopes between segments $r = 1$ to $r = 2$ and $r = 2$ to $r = 3$).

Table 10. Site Selections for the Forty-Nine City Data Set with Assignment Vector [0.6, 0.4] and $p = 15$

r	Open Sites	Closed Sites	Objective
0	1 2 3 5 7 9 14 15 20 25 31 33 35 42 43	None	6.1980856e9
1	1 2 3 5 7 9 14 15 20 25 31 33 35 42	43	9.392304e9
2	1 2 3 5 7 9 14 15 20 25 31 33 35	42 43	1.5861786e10
3	1 2 3 5 7 9 14 15 20 25 31 33	35 42 43	1.9176632e10
4	1 2 3 5 7 9 14 15 20 25 33	31 35 42 43	2.0995402e10
5	1 2 3 5 7 9 14 15 20 25	31 33 35 42 43	2.5832415e10
6	1 2 3 5 7 9 14 15 20	25 31 33 35 42 43	3.1648756e10
7	1 2 3 5 7 9 14 20	15 25 31 33 35 42 43	3.5090354e10

Note: $r = 0$, represents the no interdiction case, that is, the original vector assignment median solution for the associated assignment vector of [0.6, 0.4].

To understand the spatial interplay between different solutions, we have chosen to plot three of the patterns in Figures 2–4 taken from Tables 9 and 10. In each figure, cities are depicted as a dot along with the city names. Cities that house facilities are depicted with triangles and cities with interdicted facilities are depicted with squares. Figure 2 depicts an optimal RIVAL solution for the forty-nine city data set with the assignment vector [0.7, 0.2, 0.1] where two facilities are interdicted out of fifteen existing locations. As the figure shows, Los Angeles and Las Vegas in the Southwest are interdicted while none of the sites in the East and Midwest are interdicted. This is probably due to the fact that there is a denser cluster of existing facilities in the Northeast and interdicting a portion of the Northeast facilities does not affect weighted distance as much as interdicting two of the three facilities in the Southwest. The tables turn when the interdiction resources are increased to three as the optimal RIVAL solution for three is concentrated on the dense northeast cluster of facilities and abandons the facilities that were selected in the southwest for the $r = 1$ and $r = 2$ patterns.

Figure 3 shows a more intense interdiction involving the same data set as Figure 2, where seven facilities have been interdicted instead of two. Figure 3 shows that when interdiction resources are increased to seven, the optimal interdiction pattern is concentrated principally in the Midwest and Northeast with one city in the South (Atlanta). Virtually all existing facilities in the Northeast region are interdicted, leaving a number of high populated cities, for example, New York, served at great distances. In fact, in all the solutions for $r = 1, \ldots, 7$, interdiction either happens in the western half (for $r = 1, 2$) or in the eastern half (for $r > 2$) but never both at the same time. This makes sense because the population is somewhat divided between the East and the West. Concentrating interdiction resources on one of these two regions will effectively leave the closest open facility for a large populated region to be at a great distance away.

Figure 4 depicts the optimal interdiction pattern for $r = 2$ and $b = [0.6, 0.4]$. Note that in this example, the set of existing facilities is different than that given in Figures

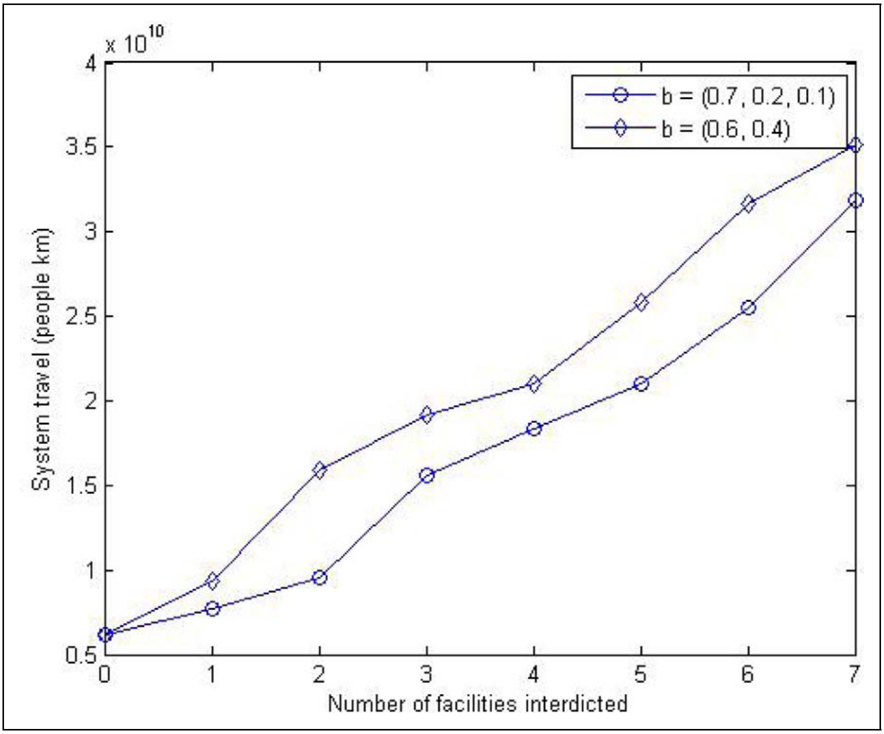

Figure 1. Total weighted distance versus number of facilities interdicted.

2 and 3 because they were based on a different assignment vector. Observe that only two facilities instead of three are located near New York in contrast with the pattern given in Figure 2. Consequently, the optimal $r = 2$ interdiction pattern is also different. With fewer existing facilities near New York, it then becomes an effective area to strike even when the resources for interdiction are low. Actually, all optimal patterns for $b = [0.6, 0.4]$ concentrate interdiction in the eastern half of the United States. The results of the RIVAL model indicate that the solutions are sensitive to the assignment vector values. Thus, it is reasonable to assume that interdiction may be sensitive to different logistics protocols as well.

Summary and Conclusions

This article presents a detailed background on constraints that have been proposed to force CA in a number of different location model settings, including budget constrained problems, competitive facilities location, obnoxious/noxious facilities location, and facility interdiction. We have also reviewed developments where

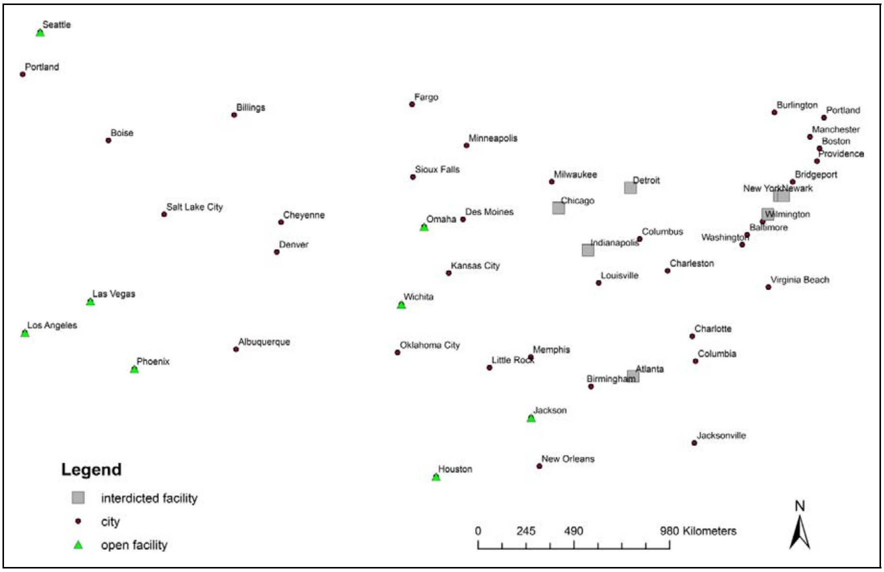

Figure 2. The RIVAL solution with $r = 2$, and assignment vector $b = [0.7,0.2,0.1]$.

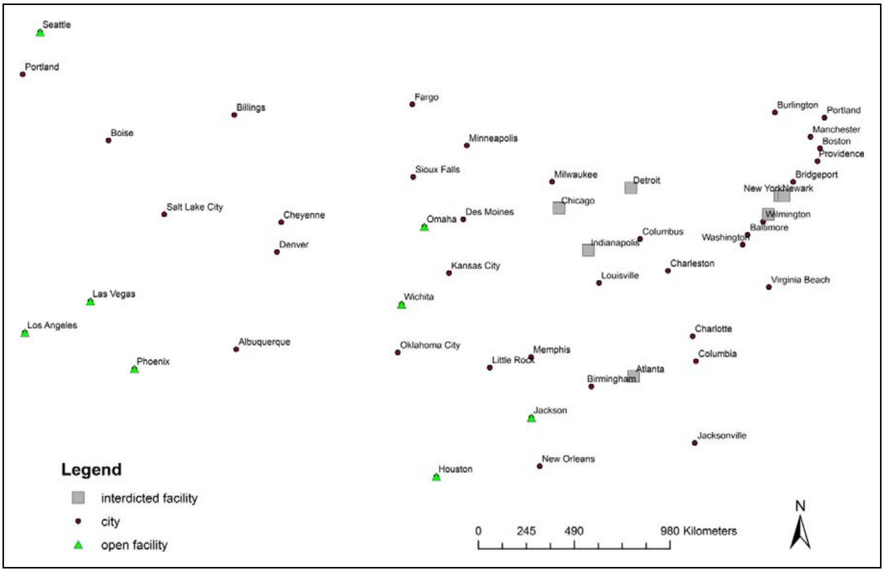

Figure 3. The RIVAL solution with $r = 7$, and assignment vector $b = [0.7, 0.2, 0.1]$.

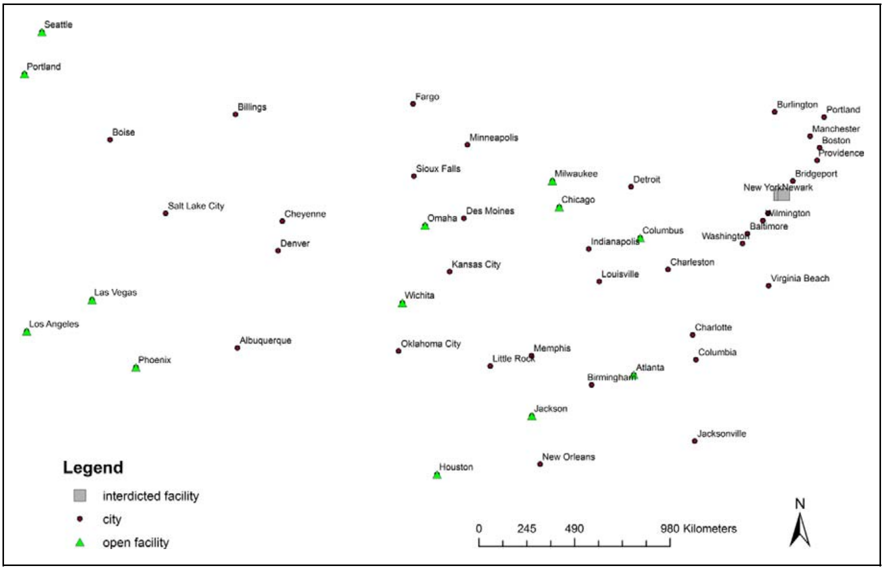

Figure 4. The optimal RIVAL solution with $r = 2$ and assignment vector of $b = [0.6, 0.4]$.

assignments to facilities other than the closest need to be considered (called multilevel assignment, in terms of the order of closest facilities). Even though there are a number of cases where multilevel ECA constraints are necessary, past work has completely overlooked this need or concentrated on specific problem cases where explicit CA constraints are not required. The main objective of this article was to show how single-level constraint structures can be generalized to a multilevel form. We also presented a new general form of a location interdiction problem which requires the use of multilevel ECA constraints (called RIVAL). We used the RIVAL model to test the efficacy of the six different generalized forms of multilevel ECA constraints. Although virtually all constraint forms, when used in conjunction with generalized Balinski constraints could be solved in a reasonable amount of computational effort, the generalized constraints based on single-level constraints of Rojeski and ReVelle (1970) and Church and Cohon (1976) are clearly more efficient in solving the RIVAL problem. We expect that the use of these constraints in other multilevel assignment/location-allocation model settings will have similar and comparable results. For the largest problems, it should be noted that the generalized form of Church and Cohon appeared to stand out as the best constraint form. Overall, we have expanded the application domain for multilevel CA location models by proposing constraints that are effective in forcing multilevel CA. There are a number of location problems (e.g., franchise, noxious, and unreliable) that should be reviewed within the context of this work and a logical next step is to apply this modeling framework in vulnerability analysis.

The RIVAL model is a new location interdiction-based construct, which can be used for analyzing worst case circumstances in the loss of one or more facilities. This model is appropriate for systems that do not adhere to CA in operation and is based on relaxing the CA assumptions of the r-interdiction problem of Church, Scaparra, and Middleton (2004).

Appendix

The Forty-Nine City Data Set

ID	X	Y	City Population	Name
1	−2,043,462.893	−290,066.7787	3,485,398	Los Angeles
2	−1,483,679.242	−474,513.6537	983,403	Phoenix
3	−1,708,838.126	−129,378.6687	258,295	Las Vegas
4	−963,415.4015	−374,720.6983	384,736	Albuquerque
5	−2,054,846.435	1,028,692.267	437,319	Portland
6	−1,331,227.857	313,885.4926	159,936	Salt Lake City
7	59,699.7336	−1,023,804.44	1,630,553	Houston
8	−1,615,855.854	691,515.5401	125,738	Boise
9	−1,965,920.403	1,243,843.692	516,259	Seattle
10	−754,974.9423	122,022.2354	467,610	Denver
11	−136,921.4531	−390,951.4457	444,719	Oklahoma City
12	−733,439.8643	273,601.9657	50,008	Cheyenne
13	588,338.6587	−971,488.5205	496,938	New Orleans
14	−117,844.8259	−144,838.7678	304,011	Wichita
15	545,728.8432	−723,760.4974	196,637	Jackson
16	−973,510.3521	821,416.347	81,151	Billings
17	332,983.5045	−468,041.1579	175,795	Little Rock
18	544,318.6218	−414,296.5325	610,337	Memphis
19	124,509.0271	14,567.1828	435,146	Kansas City
20	−1,003.7491	251,377.1036	335,795	Omaha
21	852,490.3108	−564,177.4582	265,968	Birmingham
22	−58,887.0268	504,908.5823	100,814	Sioux Falls
23	197,754.3258	288,703.4888	193,187	Des Moines
24	1,381,114.824	−852,460.7801	635,230	Jacksonville
25	1,068,336.078	−513,232.8859	394,017	Atlanta
26	−62,585.588	875,304.4698	74,111	Fargo
27	891,637.4098	−35,852.6093	269,063	Louisville
28	215,577.1279	665,426.8201	368,383	Minneapolis
29	1,386,966.571	−435,292.9495	98,052	Columbia
30	837,998.6639	131,515.3329	731,327	Indianapolis
31	686,693.8746	346,361.0187	2,783,726	Chicago
32	1,370,412.553	−307,779.8474	395,934	Charlotte
33	651,509.6186	479,965.5594	628,088	Milwaukee
34	1,244,199.831	26,132.0034	57,287	Charleston

(continued)

(continued)

ID	X	Y	City Population	Name
35	1,101,682.07	188,526.8383	632,910	Columbus
36	1,054,549.574	450,379.0274	1,027,974	Detroit
37	1,760,118.359	−57,565.1668	393,069	Virginia Beach
38	1,625,946.31	159,300.9441	606,900	Washington
39	1,650,892.104	209,552.7069	736,014	Baltimore
40	1,730,655.215	276,711.0674	71,529	Wilmington
41	1,756,665.486	313,615.8226	1,585,577	Philadelphia
42	1,817,214.993	409,929.3886	275,221	Newark
43	1,837,386.41	408,669.6128	7,322,564	New York
44	1,884,194.951	479,558.2791	141,686	Bridgeport
45	2,008,746.41	585,703.4695	160,728	Providence
46	1,791,132	834,350.9786	39,127	Burlington
47	2,021,428.629	650,526.3755	574,283	Boston
48	1,972,416.516	709,536.5859	99,567	Manchester
49	2,044,297.453	807,365.9802	64,358	Portland

Acknowledgment

The authors wish to acknowledge the support of USDOT (contract #DTOS59-07-H-0003) of this research and helpful comments from our colleagues.

Declaration of Conflicting Interests

The author(s) declared no conflicts of interest with respect to the authorship and/or publication of this article.

Funding

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article:
The authors received financial support from the USDOT (contract # DTOS59-07-H-0003) for the research.

Notes

1. Technically, a proposed ECA constraint can be found in Weaver and Church (1985); however, it was incomplete and does not apply to general multilevel assignment problems.
2. It should be recognized here that all estimates given are upper limits on the number of constraints necessary to enforce CA, as some constraints can be eliminated as not being applicable or redundant. For example, the absolute closest site to a given demand will never be a candidate for a second CA for that demand. That is, the absolute closest site will either be selected for a facility and then must be assigned to as a first closest or not selected and not assigned to. Thus, the number of constraints pertaining to the closest site will be associated with the first level, not other levels. Similar reductions are possible for the absolute second

closest site, and so on, but in general, these reductions are somewhat small in terms of the overall size of the problem for a given set size of *I*, *J*, and number of levels *L*.

3. Some of the generalized constraint forms in this section were presented by Richard Church at the 2008 NARSC Conference, New York, NY.

References

Balinski, M. L. 1965. Integer programming: Methods, uses, and computation. *Management Science Series A* 12:253-313.

Church, R. L., and J. L. Cohon. 1976. Multiobjective location analysis of regional energy facility siting problems. Report 50567, Brookhaven National Laboratory, USA.

Church, R. L., and M. P. Scaparra. 2007. Protecting critical assets: The *r*-interdiction median problem with fortification. *Geographical Analysis* 39:129-46.

Church, R. L., M. P. Scaparra, and R. Middleton. 2004. The *r*-interdiction median problem and the *r*-interdiction covering problem. *Annals of the Association of American Geographers* 94:491-502.

Church, R. L., and K. L. Roberts. 1983. Generalized coverage models and public facility location. *Papers in Regional Science* 53:117-35.

Daskin, M. S. 1983. A maximum expected covering location model: Formulation, properties and heuristic solution. *Transportation Science* 17:48-70.

Dobson, G., and U. S. Karmarkar. 1987. Competitive location on a network. *Operations Research* 35:565-74.

Gerrard, R., and R. L. Church. 1996. Closest assignment constraints and location models: Properties and structure. *Location Science* 4:251-70.

Hanjoul, P., and D. Peeters. 1987. A facility location problem with clients' preference orderings. *Regional Science and Urban Economics* 17:451-73.

Huff, D. L. 1964. Defining and estimating a trade area. *Journal of Marketing* 28:34-38.

Kolli, S., and G. Evans. 1999. A multiple objective integer programming approach for planning franchise expansion. *Computers and Industrial Engineering* 37:543-61.

Morris, J. G. 1978. On the extent to which certain fixed-charge depot location problems can be solved by LP. *Journal of the Operational Research Society* 29:71-76.

Murray, A., R. L.Church, R. Gerrard, and W. Tsui. 1998. Impact models for siting undesirable facilities. *Papers in Regional Science* 77:19-36.

Narasimhan, S., H. Pirkul, and D. Schilling. 1992. Capacitated emergency facility siting with multiple levels of backup. *Annals of Operations Research* 40: 323-37.

Pirkul, H., and D. Schilling. 1989. The capacitated maximal covering location problem with backup service. *Annals of Operations Research* 18:141-54.

ReVelle, C. S. 1993. Facility siting and integer-friendly programming. *European Journal of Operational Research* 65:147-58.

Rojeski, P., and C. S. ReVelle. 1970. Central facilities location under an investment constraint. *Geographical Analysis* 2:343-60.

Scaparra, M., and R. L. Church. 2008. A bilevel mixed-integer program for critical infrastructure protection planning. *Computers and Operations Research* 35:1905-23.

Scaparra, M. P., F. Liberatore, and M. Daskin. 2008. Analysis of facility protection strategies against uncertain numbers of attacks: The stochastic *r*-interdiction median problem with fortification. Working Paper Series, ISSN 1748-7595 No. 176.

Snyder, L., and M. Daskin. 2005. Reliability models for facility location: The expected failure cost case. *Transportation Science* 39:400-16.

Wagner, J. L., and L. M. Falkson. 1975. The optimal nodal location of public facilities with price-sensitive demand. *Geographical Analysis* 7:69-83.

Weaver, J. R., and R. L. Church. 1985. A median location model with non-closest facility service. *Transportation Science* 19:58-74.

Church R. L., and J. R. Weaver. 1986. Theoretical links between median and coverage location problems. *Annals of Operations Research* 6:1-19.