Extensions of the Psuedo-Boolean Representation of the p-Median Problem

Gabriel Sloggy
Advised by Professor Richard Church, Department of Geography

Abstract The p-median problem (PMP) is inherently a graph theoretic problem with applications in location theory and optimalization. Due to the p-median problem being NP-hard, most research in it is reduction based, generally using reduction tests and bounding, or exploring specific solvable scenarios. This paper examines and summarizes the pseudo-Boolean formulation of the problem developed by AlBdaiwi et al and applies it to a different formulation of the problem by Elloumi, creating an alternate formulation of Elloumi's model. In doing so, we demonstrate that the model developed by AlBdaiwi et al. is identical to the model developed by Cornuejols et al., albeit in a pseduo-Boolean formulation. We then provide a computational comparison of the different models. Ultimately, we offer a pseduo-Boolean formulation of the p-median problem that is more integer-friendly than that of AlBdaiwi. In comparing computational models, we show that the pseudo-Boolean Cornuejols model is slightly faster than the AlBdaiwi model, and thus assumedly, so is the pseudo-Boolean Elloumi model.

1 Introduction

The p-median problem is an NP-hard problem concerning the minimal weighted distance between a subset of points on a graph. We define the problem in terms of facilities and users as follows:

Given

- I, a set of n nodes, each of which has a particular demand
- J as set of m nodes, each of which serves as a possible facility site.
  Notice that I and J are not necessarily disjoint, and in some cases may be identical.
- $d_{ij} : I \text{x} J \to \mathbb{R}^+$, a distance function between node i and node j
- a_i, a demand or weighting variable for a demand or client i
- $p \leq m$, some subset of facilities that we want to open,

we want to determine which subset of p facilities to open in order to minimize the sum of the demand-weighted distances from each user to its nearest open facility. We assume that each facility has unlimited production capacity and that there are no restrictions on site placement.

1.1 History of the p-median Problem

The p-median problem is a generalization of a problem first posed by Fermat: given three distinct points on a plane, find a median point on the plane that minimizes the sum of the distances from each of these points to the median point. Weber (1909) generalized this to n weighted points on the plane, minimizing the sum of the weighted distance from each of these points to the median point.

The Weber problem was reformulated by Hakimi (1964) to apply to a graph by introducing the notions of the absolute center and the absolute median. Hakimi first defines a distance function $d(x, y)$ on the graph G by the length of the shortest path in G between the points x and y, where the length of a path is the sum of the weights of the branches of that path. He then defines the absolute center to be the point x_0 on an element of a weighted n-vertex of G if for all vertices v_i with weight h_i, and every point x on the graph G,

$$\max_{1\leq i\leq n} h_i d(v_i, x_0) \leq \max_{1\leq i\leq n} h_i d(v_i, x).$$

That is, the maximal weighted distance from x_0 to any vertex is less than than that of any other point on the graph.

Similarly, y_0 is defined as the absolute median if for each point y in G

$$\sum_{i=1}^{n} h_i d(v_i, y_0) \leq \sum_{i=1}^{n} h_i d(v_i, y)$$

So the sum of the weighted distances from y_0 to each of the vertices v_i is less than that of any other point on G.

Hakimi goes on to show that an absolute median of a graph is always at a vertex of a graph, thus reducing the set of possible solutions to only the vertices of the graph. In a later paper (1965), he defines a distance function between a vertex of G, v_i, and a set of p points, X_p, as $d(v_i, X_p) = \min\{d(v_i, x_1), d(v_i, x_2), ..., d(v_i, x_p)\}$ He is then finally able to define the p-median problem:
A set of points X_p^* is a p-median of G if for every X_p on G

$$\sum_{i=1}^{n} h_i d(v_i, X_p^*) \leq \sum_{i=1}^{n} h_i d(v_i, X_p)$$

The classic formulation of the p-median problem was developed by Revelle and Swain in 1970, and is given below.

The Classic formulation (RF)

$$\min Z = \sum_{i=1}^{n} \sum_{j=1}^{m} a_i d_{ij} x_{ij} \tag{1}$$

s.t.

$$\sum_{j=1}^{m} x_{ij} = 1 \text{ for } \ i = 1, 2, ..., n \tag{2}$$

$$\sum_{j=1}^{m} x_{jj} = p \tag{3}$$

$$x_{ij} \leq x_{jj} \text{ for } i = 1, 2, ..., n; \; j = 1, 2, ..., m \text{ with } i \neq j \tag{4}$$

$$x_{ij} \in \{0, 1\} \text{ for } i = 1, 2, ..., n \; j = 1, 2, ..., m. \tag{5}$$

Here, the objective function minimizes the total weighted distance accrued by assigning each demand node to its closest user facility. Constraint (2) ensures that every demand node is satisfied by exactly one facility. Constraint (3) ensures that p nodes assign themselves as the facility supplying their demand; that is, p potential sites are assigned as facilities. Constraint (4) ensures that a site i can only assign to a site j if site j is assigned as a facility. Constraint (5) indicates the binary nature of the x_{ij} variables in that facility i supplies client j or not. In particular,

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

$$x_{jj} = \begin{cases} 1 & \text{if a facility is sited at } j \text{ and meets the demand at } j \text{ as well} \\ 0 & \text{otherwise} \end{cases}$$

This model is used as the starting point for various reduction techniques, as in many of the papers mentioned below.

1.2 Reduction Techniques

Since the p-median problem is NP-hard, most research into the problem explores methods that reduce the number of variables or constraints, and hence computation time, or attempt to substitute certain steps of the problem with polynomially solvable algorithms. Various hybrid models have been suggested, such as Rosing and Revelle's heuristic concentration, a combination of a heuristic approach with linear programming using the branch-and-bound algorithm (1997). AlBdaiwi et al. (2011) analyze the problem through pseudo-Boolean polynomials, combining the work of Hammer (1968) and Beresnev (1973) in an attempt to eliminate redundant variables through an algebraic representation of the elements of the cost matrix. Church (2003) develops a model called COBRA that reduces the number of variables and constraints by eliminating redundant variables in the model. Elloumi (2010) provides a mixed-integer formulation of the problem by adapting a simple plant location problem developed by Cornuejols et a.l in 1980.

1.3 Goals

This paper summarizes the models developed by AlBdaiwi et al. and Elloumi respectively and provides a brief worked example for each of them. We then apply AlBdaiwi's pseduo-Boolean formulation to Elloumi's model, obtaining a more integer-friendly model than that of AlBdaiwi et al. but with comparable processing speed. We then compare the different models with their different reduction techniques and analyze their efficacy in reducing the problems sizes of various benchmark problems. The paper ends with conclusions and potential for future research.

2 The Pseudo-Boolean Formulation

Definition 1. *A* ***pseudo-Boolean*** *function f is a function of the form $f : \{0,1\}^n \to \mathbb{R}$ for $n \in \mathbb{N}$*

By this definition, notice that any pseudo-Boolean function f can be uniquely written as

$$f(x) = a + \sum_i a_i x_i + \sum_{i<j} a_{ij} x_i x_j + \sum_{i<j<k} a_{ijk} x_i x_j x_k + ... \tag{6}$$

The 2011 paper by AlBdaiwi et al. begins with the cost matrix of a p-median problem and manipulates it so that the objective function can then be written as a pseduo-Boolean function.
We summarize this formulation and provide a brief computational example for clarity in how the algorithm works.

2.1 The AlBdaiwi formulation

The formulation developed by AlBdaiwi et al. is penalty based. That is, the model begins at the closest site to a demand source. If the demand is not met at that site, a penalty in the form of the weighted distance between that site and the next closest site is accrued, and the next closest site is considered. This process is repeated, accummulating penalties, until a site is reached that will meet the initial site's demand. In order to formulate this concept, we must institute both an ordering based upon distance and a penalty based upon the difference in weighted distance between sites of adjacent nearness.

Let $C = [c_{ij}]$ be a cost matrix that defines the cost of supplying a demand at node i from a facility at node j.
For client i define $\Pi^i = (\pi_{ij}, ..., \pi_{nj})$ to be an ordering of $1, ..., n$ such that $c_{i\pi_{ij}} \leq c_{i\pi_{ik}}$ for all $i, k \in \{1, ..., n\}$. That is, Π^i is a permutation of $1, ..., n$ such that in row i of C, the cost in column j is less than the cost in column k if $j < k$ for all $i, k \in \{1, ..., n\}$.
We also define $\Delta^i = (\delta_{i1}, ..., \delta_{in})$ where $\delta_{i1} = c_{i\pi_{i1}}$ and $\delta_{ir} = c_{i\pi_{ir}} - c_{i\pi_{i(r-1)}}$ for $r = 2, ..., n$. That is, Δ^i is a vector of length n whose first entry is the least weighted distance from site i, and entry r being the difference in weighted distance between the r and $r-1$ farthest sites from i.
Finally, let $\mathbf{y} = (y_1, ..., y_n)$ be such that $y_i = 0$ if a plant is open at location i and 1 otherwise. So $\mathbf{y}$ is a vector representing which sites have facilities located at them.
Given this notation, we can now define the cost of satisfying a demand at i as

$$f^i(\mathbf{y}) = \delta_{i1} + \sum_{k=2}^{n} \delta_{ik} \cdot \prod_{r=1}^{k-1} y_{\pi_{ir}} \tag{7}$$

In order to aggregate the Π^i and Δ^ivectors, we define the ordering matrix $\Pi = [\pi_{ij}]$ and the difference matrix $\Delta = [\delta_{ij}]$. Notice that while Δ is distinct for a given cost matrix, Π is not necessarily so in the case that two sites have equal weighted distance from a third. With this notation, we can now represent the objective function of the p-median problem as

$$B(\mathbf{y}) = \sum_{i=1}^{n} f^i(\mathbf{y}) = \sum_{i=1}^{n} \left\{ \delta_{i1} + \sum_{k=2}^{m} \delta_{ik} \cdot \prod_{r=1}^{k-1} y_{\pi_{ir}} \right\} \tag{8}$$

But now the model is formulated strictly in terms of $y_j = 1 - x_{jj}$. We can make this substitution for x_{jj} in constraints (3) and (4) in the original formulation to form constraint (10), but we still must consider how else the constraints on the x_{ij} terms translate to constraints on the y_j terms.

To do this, when presented with a multi-part monomial $c_1 y_a y_b y_c ...$ in the objective function, we replace it with $c_1 z_k$ for some k. By doing this, we transform the objective function into a linear combination of single-part monomials. For ease of notation, we define S_k to be the set of indices of the y_i terms corresponding to z_k. For example, if $z_1 = y_1 y_2 y_4 y_7$, then $S_1 = \{1, 2, 4, 7\}$.

In application, z_k is an indicator for the t closest sites to some site j where $t = |S_k|$. So $z_k = 0$ if any y_i has a plant located at it and $z_k = 1$ if no y_j has a plant located at it for $j \in S_k$. So $z_k \geq \sum_{j \in S_k} y_j - |S_k| + 1$, which gives us constraint (11).

And thus the statement of the whole formulation is:

The AlBdaiwi formulation (AF)

$$\min\ B(\mathbf{y}) = \sum_{i=1}^{n} \left\{ \delta_{i1} + \sum_{k=2}^{m} \delta_{ik} \cdot \prod_{r=1}^{k-1} y_{\pi_{ir}} \right\} \tag{9}$$

s.t.

$$\sum_{i=1}^{n} y_i = n - p \tag{10}$$

$$\sum_{i \in S_k} y_i - z_k \leq |S_k| - 1 \text{ for each } S_k \text{ corresponding to each } z_k \tag{11}$$

$$y_i \in \{0, 1\} \text{ for } i = 1, 2, ..., n \tag{12}$$

$$z_k \geq 0 \text{ for each } k \text{ constructed.} \tag{13}$$

2.2 Reductions in the pseudo-Boolean formulation

As mentioned in the introduction, much of the research on the p-median problem focuses on eliminating variables and constraints to make runtime more efficient. This pseudo-Boolean formulation provided by AlBdaiwi et al. has two principle methods of reduction: truncation and combining like terms in the objective function.

A brief argument for truncation, also called chopping, is as follows:

Suppose we are on a network with n nodes and have to place p facilities. Consider a node i. In creating Π^i we order the sites by weighted distance from i. In the worst case scenario, all p sites are the farthest possible sites from i: those at $\Pi^i[n-p], ..., \Pi^i[n]$. i will then assign to the closest one: the site $(n-p)$ farthest from i. But in this case, i will never assign to a site farther than $(n-p)$, so we may eliminate the farthest $(p-1)$ sites from i.

The paper by Rosing, Revelle, and Rosing (1979) gives more details.

In relation to the pseudo-Boolean formulation, truncation is effected by replacing the largest p elements of row i in C with $c_{i\pi_{i(n-p+1)}}$. That is, we replace the largest p values in each column of the cost matrix C with the (p+1) largest value in the respective column.

In formulating the pseudo-Boolean function, we have instances in which the terms of a monomial are identical and we may combine them. For example, given $4y_1y_3y_6y_8 + 7y_3y_8y_1y_6$, we may combine them to get $11y_1y_3y_6y_8$. Combining similar variables in the objective function allows us the potential of removing variables and constraints, as in the given example, we effectively replaced the $y_3y_8y_1y_6$ variable with $y_1y_3y_6y_8$.

More concretely, what occurs in this scenario is that a site i has the same k closest sites as a site j. Then the decision of where to assign a facility is the same for j as it is for i, and we may substitute the corresponding variables of i in for j. In order to illustrate the process of forming the ordering and difference matrices, as well as the reduction process, the following computational example is given. This example does not reduce by truncation, as such a reduction is not unique to this formulation, and may confuse the process.

2.3 A computational example

Consider a p-median problem with $m = 5$ potential facility sites, $n = 5$ clients and $p = 2$ facilities to place, with cost matrix

$$C = \begin{pmatrix} 5 & 7 & 3 & 9 & 12 \\ 4 & 2 & 1 & 4 & 7 \\ 11 & 7 & 4 & 8 & 10 \\ 7 & 5 & 9 & 2 & 2 \\ 10 & 6 & 9 & 3 & 4 \end{pmatrix}.$$

A possible ordering matrix for C is

$$\Pi = \begin{pmatrix} 3 & 1 & 2 & 4 & 5 \\ 3 & 2 & 1 & 4 & 5 \\ 3 & 2 & 4 & 5 & 1 \\ 5 & 4 & 2 & 1 & 3 \\ 4 & 5 & 2 & 3 & 1 \end{pmatrix}$$

which has corresponding difference matrix

$$\Delta = \begin{pmatrix} 3 & 2 & 2 & 2 & 3 \\ 1 & 1 & 2 & 0 & 3 \\ 4 & 3 & 1 & 2 & 1 \\ 2 & 0 & 3 & 2 & 2 \\ 3 & 1 & 2 & 3 & 1 \end{pmatrix}.$$

We can then formulate the objective function in terms of the pseudo-Boolean polynomial:

$$\begin{aligned} B(\mathbf{y}) = 3 &+ 2y_3 + 2y_1y_3 + 2y_1y_2y_3 + 3y_1y_2y_3y_4 \\ &+ 1 + 1y_3 + 2y_1y_3 + 0y_1y_2y_3 + 3y_1y_2y_3y_4 \\ &+ 4 + 3y_3 + 1y_2y_3 + 2y_2y_3y_4 + 1y_2y_3y_4y_5 \end{aligned}$$

$$+ 2 + 0y_5 + 3y_4y_5 + 2y_2y_4y_5 + 2y_1y_2y_4y_5$$
$$+ 3 + 1y_4 + 2y_4y_5 + 3y_2y_4y_5 + 1y_2y_3y_4y_5$$

Combining similar terms, we get

$$\begin{aligned}B(\mathbf{y}) &= 13 + 6y_3 + 1y_4 + 4y_1y_3 + 1y_2y_3 + 5y_4y_5 + 2y_1y_2y_3\\ &+ 2y_2y_3y_4 + 5y_2y_4y_5 + 6y_1y_2y_3y_4 + 2y_1y_2y_4y_5 + 2y_2y_3y_4y_5\\ &= 13 + 6y_3 + 1y_4 + 4z_1 + 1z_2 + 5z_3 + 2z_4 + 2z_5 + 5z_6 + 6z_7 + 2z_8 + 2z_9\end{aligned}$$

Notice that while the original objective function had 23 non-zero monomials, the reduced objective function has only 12. We can now formulate the model as:

$$\begin{aligned}\min\ & B(\mathbf{y}) = 13 + 6y_3 + 1y_4 + 4z_1 + 1z_2 + 5z_3 + 2z_4 + 2z_5 + 5z_6 + 6z_7 + 2z_8 + 2z_9\\ \text{s.t.}\ & \\ & \sum_{j=1}^{5} y_j = 3\\ & \sum_{j\in S_k} y_j - z_k \leq |S_k| - 1 \text{ for each } S_k \text{ corresponding to each } z_k\\ & y_j \in \{0,1\} \text{ for } j = 1, 2, ..., 5\\ & z_k \geq 0 \text{ for each } k \text{ constructed.}\end{aligned}$$

3 The Elloumi formulation

The formulation of the p-median problem developed by Elloumi (2008) is also a penalty-based formulation, and as we will see, is very similar to the pseudo-Boolean model developed by AlBdaiwi et al. The Elloumi model is based upon a formulation developed by Cornuejols et al. in 1980, which we will show is identical to the AlBdaiwi formulation when translated into a pseudo-Boolean form. For ease of notation, hereafter we will refer to the different formulations by their initials: the classical model developed by Revelle and Swain is RF, the pseudo-Boolean model developed by AlBdaiwi et al. is AF, Elloumi's formulation is EF, and Cornuejol et al.'s formulation is CF. We now introduce notation common to CF and EF.

For a site i, let K_i be the number of differrent distances from site i to any facility. Notice that K_i does not necessarily equal p as some facilities may be equidistant from i, so $K_i \leq p$. Let $D_i^1 < D_i^2 < ... < D_i^{K_i}$ be these distances in increasing order.
Let $V_i^k = \{j : d_{ij} \leq D_i^k\}$. That is, V_i^k is the neighborhood composed of the k closest facilities to i, for $k = 1, 2, ..., K_i$.
With these definitions, V_i^1 is the equidistant set of sites closest to i, and $V_i^{K_i}$ is the set of all facility sites. Then in an optimal solution, a client i will assign to the smallest neighborhood V_i^k containing an open facility.

3.1 The Cornuejols formulation

In formulating CF, Cornuejols et al. define y_j equal to the x_{jj} variables used in RF. That is, $y_j = 1$ if a facility is open and 0 if closed. Notice that this definition of y_j is opposite that of the AF definition of y_j. While it is confusing, we intentionally keep this discrepancy, as its resolution identifies the two models. The CF also introduces binary variables of the form z. In particular, $z_i^k = 1$ if and only if every facility in V_i^k is closed, and 0 otherwise. Alternately, $z_i^k = 0$ if any facility in V_i^k is open, and 1 otherwise. Using these z_i^k and y_i we can now present Cornuejols et al.'s formulation of the p-median problem.

The Cornuejols formulation (CF)

$$\min Z(z,y) = \sum_{i=1}^{n} \left\{ D_i^1 + \sum_{k=1}^{K_i-1} (D_i^{k+1} - D_i^k) z_i^k \right\} \tag{14}$$

s.t.

$$\sum_{j=1}^{m} y_j = p \tag{15}$$

$$z_i^k + \sum_{j:d_{ij} \leq D_i^k} y_j \geq 1 \text{ for } \; i = 1,2,...,n; \; k = 1,2,...,K_i \tag{16}$$

$$z_i^{K_i} = 0 \text{ for } i = 1,2,...,n \tag{17}$$

$$z_i^k \geq 0 \text{ for } i = 1,2,...,n; \; k = 1,2,...,K_i \tag{18}$$

$$y_j \in \{0,1\} \text{ for } j = 1,2,...,m \tag{19}$$

Constraint (15) is the same as constraint (3) in RF; that is, it ensures that exactly p sites will assign facilities to be opened. Constraint (16) ensures that for a given client i, either $z_i^k = 1$ or at least one facilitiy in V_i^k is open. Constraint (17) ensures that for a given client i, at least one facility in the whole neighborhood $V_i^{K_i}$ is open. This constraint is redundant, as given constraint (15), p sites will be assigned, so $z_i^{K_i}$ will certainly be 0. We will heretofore ignore this contraint in our own developments of the model. Constraints (18) and (19) are positivity and binary constraints for the respective variables. The objective function (14) sums the total weighted distance from each client i to the facility site chosen to provide for it.

Remark 1. *AF and CF are functionally identical.*

We restate the formulation of AlBdaiwi et al. for ease of comparison:

The AlBdaiwi formulation (AF)

$$\min B(\mathbf{y}) = \sum_{i=1}^{n} \left\{ \delta_{i1} + \sum_{k=2}^{m} \delta_{ik} \cdot \prod_{r=1}^{k-1} y_{\pi_{ir}} \right\}$$

s.t.

$$\sum_{i=1}^{n} y_i = n - p$$

$$\sum_{i \in S_k} y_i - z_k \leq |S_k| - 1 \text{ for each } S_k \text{ corresponding to each } z_k$$

$y_i \in \{0,1\}$ for $j = 1,2,...,n$
$z_k \geq 0$ for each k constructed.

The immediate difference in the two models is that they have switched the definitions of i and j. That is, in the AlBdaiwi et al. model, i represents facilities and j demand, while in most other models i represents demand and j facilities. To mitigate this, we present an altered formulation of CF, using the AF definition of y_j. Hence we replace every y_j in CF with $\bar{y}_j = 1 - y_j$ and z_i^k with $\bar{z}_i^k = 1 - z_i^k$.

The altered Cornuejols formulation (CF')

$$\min\ Z(z,y) = \sum_{i=1}^{n} \left\{ D_i^1 + \sum_{k=1}^{K_i-1} (D_i^{k+1} - D_i^k) z_i^k \right\} \tag{20}$$

s.t.

$$\sum_{j=m}^{n} \bar{y}_j = m - p \tag{21}$$

$$\sum_{j:d_{ij} \leq D_i^k} \bar{y}_j - z_i^k \leq |D_i^k| - 1 \text{ for } i = 1,2,...,n;\ k = 1,2,...,K_i \tag{22}$$

$$\bar{z_i}^{K_i} = 1 \text{ for } i = 1,2,...,n \tag{23}$$

$$\bar{y}_j \in \{0,1\} \text{ for } j = 1,2,...,m \tag{24}$$

$$\bar{z_i}^k \geq 0 \text{ for } i = 1,2,...,n;\ k = 1,2,...,K_i \tag{25}$$

Computations:
Let $\bar{y}_j = 1 - y_j, \bar{z}_i^k = 1 - z_i^k$

(21) $\sum_{j=1}^{m} \bar{y}_j = \sum_{j=1}^{m} (1 - y_j) = m - \sum_{j=1}^{m} y_j = m - p$

(22) $z_i^k + \sum_{j:d_{ij} \leq D_i^k} y_j \geq 1$

$\iff z_i^k + \sum_{j:d_{ij} \leq D_i^k} (1 - \bar{y}_j) \geq 1$

$\iff z_i^k + |D_i^k| - \sum_{j:d_{ij} \leq D_i^k} \bar{y}_j \geq 1$

$\iff z_i^k - \sum_{j:d_{ij} \leq D_i^k} \bar{y}_j \geq 1 - |D_i^k|$

$\iff \sum_{j:d_{ij} \leq D_i^k} \bar{y}_j - z_i^k \leq |D_i^k| - 1$

(23) $z_i^{K_i} = 0$
$\iff 1 - z_i^{K_i} = 1$
$\iff \bar{z_i}^{K_i} = 1$
(24) This follows immediately because by definition $y_i \in \{0,1\} \Rightarrow \bar{y}_i \in \{0,1\}$
(25) $\bar{z}_i^k \geq 0$ because by definition $z_i^k \in \{0,1\} \Rightarrow \bar{z}_i^k \in \{0,1\}$

We leave it to the reader to determine that $D_i^1 = \delta_{i1}$, that for $k > 1, (D_i^{k+1} - D_i^k) = \delta_{ik}$, and that $z_i^k = y_{\pi_{ir}}$. The results follow from examining the definitions of the relevant terms.

Upon this substitution, we see that the models are nearly identical, with the exception being the extra constraint (23) in CF'.

3.2 The Elloumi formulation

The formulation that Elloumi develops is nearly identical to CF, except that constraint (16) is separated into two distinct contraints based upon a recursive definition of z. Notice how this effects the summation in each of the relevant constraints: in CF, constraint (16) sums over all nodes within distance k or less, while in EF, constraints (28) and (29) sum over only those nodes at the exact distance of k. EF effectively adds another set of constraints in order to make the constraints smaller and easier to manipulate. EF uses the exact same notation as CF; the following recursive statement is manipulation not definition.

$$z_i^k = \prod_{j \in V_i^k} (1 - y_j), \text{ for } i = 1, 2, ..., n; \ k = 1, 2, ..., K_i$$

which implies the following recursive definition:

$$z_i^1 = \prod_{j:d_{ij}=D_i^1} (1 - y_j), \text{ for } i = 1, ..., n$$
$$z_i^k = z_i^{k-1} \prod_{j:d_{ij}=D_i^k} (1 - y_j), \text{ for } i = 1, ..., n; \ k = 2, 3, ..., K_i.$$

This motivates the formulation of the Elloumi model:

The Elloumi formulation (EF)

$$\min Z(z, y) = \sum_{i=1}^{n} \left\{ D_i^1 + \sum_{k=1}^{K_i - 1} (D_i^{k+1} - D_i^k) z_i^k \right\} \tag{26}$$

s.t.

$$\sum_{j=1}^{m} y_j = p \tag{27}$$

$$z_i^1 + \sum_{j:d_{ij}=D_i^1} y_j \geq 1 \text{ for } \ i = 1, 2, ..., n \tag{28}$$

$$z_i^k + \sum_{j:d_{ij}=D_i^k} y_j \geq z_i^{k-1} \text{ for } \ i = 1, 2, ..., n; \ k = 1, 2, ..., K_i \tag{29}$$

$$z_i^{K_i} = 0 \text{ for } i = 1, 2, ..., n \tag{30}$$

$$z_i^k \geq 0 \text{ for } i = 1, 2, ..., n; \ k = 1, 2, ..., K_i \tag{31}$$

$$y_j \in \{0, 1\} \text{ for } j = 1, 2, ..., m \tag{32}$$

As mentioned previously, the only difference between CF and EF is that constraint (16) in CF is broken up into constraints (28) and (29) in EF. Another way of conceptualizing this is to consider the sets over which the summations occur. CF has a filled ring surrounding the relevant z_j containing entries of distance D_i^k or less; while EF has a hollow ring surrounding the relevant z_j, only containing entries precisely D_i^k away.

Elloumi goes on to show that if $\underline{\text{CF}}$ and $\underline{\text{EF}}$ are the LP-relaxations of CF and EF respectively, then

1. The feasible solution set of EF is included in the feasible solution set of CF, with the inclusion possibly being strict.
2. CF and EF have the same optimal values.
3. Given an optimal solution to one of CF or EF, we can deduce an optimal solution of the other with the same optimal value.

So the Elloumi model is at least equivalent to the Cornuejols model, with the possible advantage of having a tighter feasible region.

3.3 Reductions in the Elloumi formulation

Elloumi offers three reduction steps inherent in the definition of the z variables, though one is trivial, so only two are presented.

1. For any client i, if V_i^1 is a singleton y_a then $z_i^1 = 1 - y_a$ for any feasible solution. Then the variable z_i^1 can be replaced by $1 - y_a$, and the constraint defining z_i^1, $z_i^1 + y_a \geq 1$ may be eliminated. In application, for a client i, if there is only one site the closest distance away, then we may replace the z variable representing that site with $1 - y_a$ and eliminate its corresponding constraint.
2. For any two clients i and i', if $V_i^k = V_{i'}^{k'}$ for some k, k', then $z_i^k = z_{i'}^{k'}$ for any feasible solution. We may then replace the variable $z_{i'}^{k'}$ with z_i^k, and eliminate the constraint defining $z_{i'}^{k'}$: $z_{i'}^{k'} + \sum_{j:di'j=D_{i'}^{k'}} y_j \geq z_{i'}^{k'-1}$. That is, if given two clients i and i', if the sets of the k and k' respective closest facility sites are identical, then we may replace $z_{i'}^{k'}$ with z_i^k and eliminate the constraint representing $z_{i'}^{k'}$. Notice that this reduction is identical to the combining of terms that occurs in AF.

3.4 A computational example

We use the same scenario and cost matrix as with the AlBdaiwi model to illustrate the similarities and differences between the differrent formulations.

Consider a p-median problem with $m = 5$ potential facility sites, $n = 5$ clients and $p = 2$ facilities to place, with cost matrix

$$C = \begin{pmatrix} 5 & 7 & 3 & 9 & 12 \\ 4 & 2 & 1 & 4 & 7 \\ 11 & 7 & 4 & 8 & 10 \\ 7 & 5 & 9 & 2 & 2 \\ 10 & 6 & 9 & 3 & 4 \end{pmatrix}.$$

Let each column represent a facility F_i and each row a client C_i for $i = 1, 2, ..., 5$.

$V_1^1 = \{F_3\}$	$V_1^2 = \{F_1, F_3\}$	$V_1^3 = \{F_1, F_3, F_4\}$	$V_1^4 = \{F_1, F_2, F_3, F_4\}$	$V_1^5 = V_i^{K_i}$
$V_2^1 = \{F_3\}$	$V_2^2 = \{F_2, F_3\}$	$V_2^3 = \{F_2, F_3, F_4\}$	$V_2^4 = \{F_1, F_2, F_3, F_4\}$	$V_2^5 = V_i^{K_i}$
$V_3^1 = \{F_3\}$	$V_3^2 = \{F_3, F_4\}$	$V_3^3 = \{F_2, F_3, F_4\}$	$V_3^4 = \{F_2, F_3, F_4, F_5\}$	$V_3^5 = V_i^{K_i}$
$V_4^1 = \{F_4, F_5\}$	$V_4^2 = \{F_2, F_4, F_5\}$	$V_3^4 = \{F_2, F_3, F_4, F_5\}$	$V_4^4 = V_i^{K_i}$	
$V_5^1 = \{F_4\}$	$V_5^2 = \{F_4, F_5\}$	$V_3^5 = \{F_2, F_4, F_5\}$	$V_5^4 = \{F_2, F_3, F_4, F_5\}$	$V_5^5 = V_i^{K_i}$

Table 1: Neighborhoods of the computational example

We can now formulate the objective function of the opimization problem:

$$\begin{aligned} Z(z,y) = 3 &+ 2z_1^1 + 2z_1^2 + 2z_1^3 + 3z_1^4 \\ &+ 1 + 1z_2^1 + 2z_2^2 + 3z_2^3 \\ &+ 4 + 3z_3^1 + 1z_3^2 + 2z_3^3 + 1z_3^4 \\ &+ 2 + 3z_4^1 + 2z_4^2 + 2z_4^3 \\ &+ 3 + 1z_5^1 + 2z_5^2 + 3z_5^3 + 1z_5^4 \\ \\ &= 13 + 2z_1^1 + 2z_1^2 + 2z_1^3 + 3z_1^4 + 1z_2^1 + 2z_2^2 + 3z_2^3 + 3z_3^1 + 1z_3^2 \\ &+ 2z_3^3 + 1z_3^4 + 3z_4^1 + 2z_4^2 + 2z_4^3 + 1z_5^1 + 2z_5^2 + 3z_5^3 + 1z_5^4 \end{aligned}$$

Then the complete model is:

$$\begin{aligned} \min\ Z(z,y) = 13 &+ 2z_1^1 + 2z_1^2 + 2z_1^3 + 3z_1^4 + 1z_2^1 + 2z_2^2 + 3z_2^3 + 3z_3^1 + 1z_3^2 \\ &+ 2z_3^3 + 1z_3^4 + 3z_4^1 + 2z_4^2 + 2z_4^3 + 1z_5^1 + 2z_5^2 + 3z_5^3 + 1z_5^4 \\ \text{s.t.} \\ &\sum_{j=1}^{5} y_j = 2 \\ &z_i^1 + \sum_{j:d_{ij}=D_i^1} y_j \geq 1 \text{ for } \ i = 1,2,...,5 \\ &z_i^k + \sum_{j:d_{ij}=D_i^k} y_j \geq z_i^{k-1} \text{ for } \ i = 1,2,...,5; \ k = 1,2,...,K_i \\ &z_i^{K_i} = 0 \text{ for } i = 1,2,...,5 \\ &z_i^k \geq 0 \text{ for } i = 1,2,...,5; \ k = 1,2,...,K_i \\ &y_j \in \{0,1\} \text{ for } j = 1,2,...,5 \end{aligned}$$

4 A pseudo-Boolean representation of the Elloumi model

Having discovered that the pseudo-Boolean formulation developed by AlBdaiwi et al. is functionally equivalent to the formulation presented by Cornuejols et al., we are motivated to present a pseudo-Boolean formulation of the model presented by Elloumi. In the paper introducing Elloumi's model, EF ran significantly faster than both CF and RF solving a variety of benchmark p-median problems from ORLIB (Beasley 1990) and rw (Resende and Werneck 2004). This leads us to believe that a pseudo-Boolean formulation of EF will run faster than CF's pseudo-Boolean counterpart, AF.

We set about converting the pseudo-Boolean AF into a pseudo-Boolean formulation following the same set of constraints. We begin with the AlBdaiwi formulation and make the substitution $x_j = (1 - y_j)$. Remember that y_j in AF is $\bar{y_j}$ in all other formulations, so x_j will equal y_j outside of AF.

Making this substitution, the original AF

The AlBdaiwi formulation (AF)

$$\min B(\mathbf{y}) = \sum_{i=1}^{n} \left\{ \delta_{i1} + \sum_{k=2}^{m} \delta_{ik} \cdot \prod_{r=1}^{k-1} y_{\pi_{ir}} \right\}$$

s.t.

$$\sum_{i=1}^{n} y_i = n - p$$

$$\sum_{i \in S_k} y_i - z_k \leq |S_k| - 1 \text{ for each } S_k \text{ corresponding to each } z_k$$

$$y_i \in \{0, 1\} \text{ for } j = 1, 2, ..., n$$

$$z_k \geq 0 \text{ for each } k \text{ constructed.}$$

becomes

The altered AlBdaiwi formulation

$$\min B(\mathbf{y}) = \sum_{i=1}^{n} \left\{ \delta_{i1} + \sum_{k=2}^{m} \delta_{ik} \cdot \prod_{r=1}^{k-1} (1 - x_{\pi_{ir}}) \right\} \tag{33}$$

s.t.

$$\sum_{j=1}^{n} x_j = p \tag{34}$$

$$\sum_{j \in S_k} x_j + z_k \geq 1 \text{ for each } S_k \text{ corresponding to each } z_k \tag{35}$$

$$x_j \in \{0, 1\} \text{ for } j = 1, 2, ..., m \tag{36}$$

$$z_k \geq 0 \text{ for each } k \text{ constructed.} \tag{37}$$

Calculations

Constraints (34), (36), and (37), as well as the objective function (33) are identical to the calculations performed in substituting $\bar{y_j}$ for y_j in CF', for (21), (24), (25), and (20) respectively.

Constraint (35) is simple but not necessarily apparent.

$$\sum_{j \in S_k} y_j - z_k \leq |S_k| - 1$$

$$\implies \sum_{j \in S_k} (1 - x_j) - z_k \leq |S_k| - 1$$

$$\implies |S_k| - \sum_{j \in S_k} x_j - z_k \leq |S_k| - 1$$
$$\implies \sum_{j \in S_k} x_j + z_k \geq 1$$

At this point, the altered AF is beginning to resemble EF. All that remains is to convert constraint (35):

$$\sum_{j \in S_k} x_j + z_k \geq 1 \text{ for each } S_k \text{ corresponding to each } z_k$$

We recall the definitions of S_k and D_i^k. In AF, for a multi-part monomial $y_a y_b y_c ... = z_k$, S_k is the set of $a, b, c, ...$ corresponding to the indices of the variables making up z_k. In application, S_k represents the t closest sites to a site j where $t = |S_k|$.
D_i^k is the ordered set of distances of the k closest sites to a site i.
In order to make constraint (35) consistent with the notation of EF, we change our notation from S_k to D_i^k. We want to preserve the contents of the set, so $\{j : j \in S_k\} = \{j : d_{ij} \leq D_i^k\}$. But in doing this, we also must change z_k to z_i^k.
We now can reformulate constraint (35) as

$$\sum_{j : d_{ij} \leq D_i^k} x_j + z_i^k \geq 1 \text{ for } i = 1, 2, ..., n$$

This essentially gives us the formulation of Cornuejols et al.

The pseudo-Boolean Cornuejols model (PBC)

$$\min Z(y) = \sum_{i=1}^{n} \left\{ \delta_{i1} + \sum_{k=2}^{m} \delta_{ik} \cdot \prod_{r=1}^{k-1} (1 - x_{\pi_{ir}}) \right\} \tag{38}$$

s.t.

$$\sum_{j=1}^{m} x_j = p \tag{39}$$

$$\sum_{j : d_{ij} \leq D_i^k} x_j + z_i^k \geq 1 \text{ for } i = 1, 2, ..., n \tag{40}$$

$$x_j \in \{0, 1\} \text{ for } j = 1, 2, ..., m \tag{41}$$

$$z_k \geq 0 \text{ for each } k \text{ constructed.} \tag{42}$$

By splitting constraint (35) into two new constraints, we now have a pseudo-Boolean formulation of Elloumi's model:

The pseudo-Boolean Elloumi model (PBE)

$$\min Z(y) = \sum_{i=1}^{n} \left\{ \delta_{i1} + \sum_{k=2}^{m} \delta_{ik} \cdot \prod_{r=1}^{k-1} (1 - x_{\pi_{ir}}) \right\} \tag{43}$$

s.t.

$$\sum_{j=1}^{m} x_j = p \tag{44}$$

$$z_i^1 + \sum_{j:d_{ij}=D_i^1} x_j \geq 1 \text{ for } i = 1, 2, ..., n \tag{45}$$

$$z_i^k + \sum_{j:d_{ij}=D_i^k} x_j \geq z_i^{k-1} \text{ for } i = 1, 2, ..., n \tag{46}$$

$$x_j \in \{0, 1\} \text{ for } j = 1, 2, ..., m \tag{47}$$

$$z_k \geq 0 \text{ for each } k \text{ constructed.} \tag{48}$$

This PBE model essentially uses the pseudo-Boolean objective function of AF and the constraints of EF or CF. We may also apply our choice of reduction techniques from the two formulations: while they ultimately end in the same result, one reduction may be preferred over the other in a given situation. For example, depending on the modeling language, the AF reductions may be easier because they merely involve polynomial algebra, while the EF reductions rely on set content comparison. Thus the primary benefit to hybridizing these two models is the retention of the pseudo-Boolean objective function from AF and the flexibility of reduction options from both models.

Though the sets of constraints are nearly identical between AF and EF, using the constraints of EF provides one small advantage compared to AF: constraints (45) and (46) in the PBE are more integer-friendly than constraint (11). Compare

$$z_i^1 + \textstyle\sum_{j:d_{ij}\leq D_i^1} x_j \geq 1 \text{ for } i = 1, 2, ..., n \quad (45)$$
$$z_i^k + \textstyle\sum_{j:d_{ij}\leq D_i^k} x_j \geq z_i^{k-1} \text{ for } i = 1, 2, ..., n \quad (46)$$

with the AF constraint

$$\sum_{i\in S_k} y_i - z_k \leq |S_k| - 1 \text{ for each } S_k \text{ corresponding to each } z_k \quad (11)$$

Revelle (1993) discusses how the formulation of a model affects the nature of its solution, particularly whether the solution is integer or fractional. He concludes that, generally, a solution is more likely to be integer if the coefficients of the variables in the constraint set are -1, 0, or 1. He refers to this property as being integer-friendly In the PBE constraints (45) and (46), we have each of our variables and our bound taking values of 0 or 1. In the AF constraint (11), we have the variables taking values of 0 or 1, but the bounding constraint can take a value up to $n - 1$ in size. While there may be some unexplored repercussion of having an unbounded greater-than inequality, what with the lesser-than inequality bounded below by zero, Revelle's notion of integer-friendliness makes the PBE or CBE formulation of the constraints preferable for now.

5 Computational Comparison

For the computational aspect of this paper, we developed models for the AF and PBC models. We assumed that the PBE model would be faster than the PBC model, so if we can show that the PBC model is faster than the AF model, we assume that the PBE model is faster than the AF model as well. Comparing the PBC and AF models is tenuous, as they are nearly identical with the exception of the PBC model's integer friendly set of constraints.

As such, we expect only marginal advantages in run time.

To best isolate the effect that the constraint formulation has on model speed, we refrain from performing other reductions on these models. Thus the runtimes provided and variables and constraints eliminated are not lower bounds: the results of the PBC model are intended solely to be taken in context of comparison, not as a benchmark.

Using Code::Blocks C++ as our modeling language, we have developed algorithms to perform the reductions appropriate to each formulation and to formulate the AlBdaiwi and hybrid models. We run our models against several problem datasets from the Beasley OR library, varying in their sizes and values of p. Preprocessing occurred on a portable PC with a 2.00 GHz Intel Core 2 Duo processor and 3.00 GB of RAM. The models were run on a non-portable PC with a 2.13 GHz Intel Core 2 Duo processor and 2.00 GB of RAM. The results are solved in Xpress IVE (FICO), courtesy of the UCSB Department of Geography. We then compare the reduction efficiency, preprocessing times, and solving times of each model.

5.1 Discussion of Models

In this section, we focus exclusively on AlBdaiwi et al.'s pseudo-Boolean mode and the pseudo-Boolean Cornuejols model developed in this paper. The classic model developed by Revelle has been thoroughly tested already, and in their respective papers, AlBdaiwi et al. and Elloumi both show their respective models to be faster in regard to certain benchmark problems. Similarly, Elloumi shows her model to be more efficient than that of Cornuejols et al. For these reasons, we choose to focus on AF and PBC, as we assume that the PBE is faster than the PBC model.

In running these models, we seek to investigate the effects of the PBE model's integer friendly constraints, as opposed to the non-integer friendly constraints of the AF model. For this reason, we refrain from including any of the aforementioned reduction methods in order to isolate the effects of the different constraint formulations on the model run time.

5.2 Computational Results

The Beasley library from which we have drawn these problems has n=m; that is, the problem treats each node as a potential facility site and a potential client. Thus, the problem size, number of demand sites, and number of facility sites are all the same for the pmed problems. Similarly, notice that the number of entries in the cost matrix is equal to n^2.

The percent variable reduction is given by (n^2−remaining variables)/n^2. Notice that since the AF and PBE models use the same objective function, the number of remaining variables and variable reduction are the same for each model. The preprocessing and run times are rounded to the nearest hundredth of a second.

In the AF model, we see that preprocessing time increases exponentially as n increases. The run times for this model fluctuate broadly, with little reason immediately apparent. We attribute it to differences in the complexity of the problem datasets, as the problems from

	Size (n)	p	Var. after	Var. red. (%)	**AF** Preproc. (sec)	**AF** Run time (sec)	**PBC** Preproc. (sec)	**PBC** Run time (sec)
pmed1	100	5	7407	25.93	3.19	1.45	3.14	1.37
pmed2	100	10	7633	23.67	3.38	3.26	3.36	2.05
pmed3	100	10	7472	25.28	3.33	3.36	3.34	2.70
pmed4	100	20	7472	25.28	3.44	1.34	3.40	1.39
pmed5	100	33	7226	27.74	3.30	1.16	3.31	1.16
pmed6	200	5	16508	58.73	26.57	28.77	26.48	28.48
pmed7	200	10	16302	59.25	27.15	6.89	26.18	6.73
pmed8	200	20	16868	57.83	27.22	5.82	27.96	5.79
pmed9	200	40	16274	59.32	26.66	5.32	27.36	5.94
pmed10	200	67	14471	63.82	25.63	4.38	26.03	4.61
pmed11	300	5	20195	77.56	94.68	24.43	91.34	16.63
pmed12	300	10	21083	76.57	93.27	45.91	90.33	26.56
pmed13	300	30	21281	76.35	95.75	46.04	92.73	11.81
pmed14	300	60	22091	75.45	96.02	11.31	93.92	11.06
pmed15	300	100	19882	77.91	91.66	7.97	90.33	7.71
pmed16	400	5	23460	85.33	235.08	70.01	238.10	66.94
pmed17	400	10	22784	85.76	233.175	69.89	232.00	18.44
pmed18	400	40	24984	84.39	240.44	19.78	237.14	13.88
pmed19	400	80	23336	85.42	235.71	14.18	231.87	14.77
pmed20	400	133	24630	84.61	240.00	14.91	235.96	14.21
pmed21	500	5	25648	89.74	834.07	28.58	496.19	26.04
pmed22	500	10	27800	88.89	240.00	24.35	502.95	23.60
pmed23	500	50	27340	89.06	498.92	22.03	493.54	21.85
pmed24	500	100	26740	89.30	498.75	20.52	494.12	20.36
pmed25	500	167	27231	89.11	497.32	21.74	491.37	20.842

Table 2: Reductions of monomials of the objective function as well as preprocessing and run times of the AF and PBC models for benchmark problems

the Beasley library are all distinct. That is, they do not use the same dataset for each n and vary p, but rather use a different dataset for each instance.

In the PBE model,the preprocessing times are comparable to the AF model. There are a few discrepancies either way, as with pmed21 and pmed22, but we attribute these to anomalies of the computer. Like the AF model, the run time for the PBE model does not correlate smoothly with the size of the dataset. Again, we attribute these discrepancies to the irregularity of the Beasley datasets. Comparing the run time results to the AF model, we see that they are just slightly faster for n up to 500. While we might normally dismiss this as a computational anamoly, its consistency makes an argument for it.

In particular, we see that in all but three of the instances, the PBC model runs faster than the AF model. In some cases, such as pmed13 and pmed17, PBC runs over three times as fast as AF. Averaging the difference in times for each of these simulations, PBC was over 5 seconds faster than AF. Taking the average with differences greater than three seconds removed, we see that PBC ran .48 seconds faster than AF. As we had predicted initially, this indicates that the integer-friendly constraint (45) of the PBC model has a slight advantage over the normal constraint (11) of the AF model, at least in the solving software Xpress IVE.

On the whole, the PBE model performed better than the AF model in both preprocessing and run time. The advantage in preprocessing is unexpected and unattributed. The advantage in run time is marginal, but present. The models could be run on larger values of n to see if the advantage held by the PBC model widens as n increases.

6 Conclusion

This paper begins with an overview of the p-median problem, its history, and the classic formulation by Revelle and Swain (1970). It then gives a brief explanation of the development of the respective models developed by AlBdaiwi et al. and Elloumi, presents the models, explains the reduction techniques inherent in each model, and provides a worked example of a sample problem for each. We then set about converting Elloumi's model into a pseudo-Boolean form.

In this conversion process, we determine that the model developed by Cornuejols et al. (1980), a predecessor of Elloumi's model, is functionally equivalent to AlBdaiwi et al.'s model. Upon a particular variable substitution and ignoring extraneous constraints, we can see that the two models are equivalent.

This motivated us to develop a pseudo-Boolean formulation of Elloumi's model. In Elloumi's paper featuring her model, that model had proved more efficient than the model developed by Cornuejols et al. This led to the actual development of models for the AlBdaiwi et al. and pseudo-Boolean Cornuejols formulations, operating under the assumption that the pseudo-Boolean Elloumi model would outperform the PBC model, and hence the AF model.

In formulating the hybrid model, we essentially ended with the objective function from the AlBdaiwi et al. model and the constraints of the Cornuejols model This allowed us some modicum of flexibility in choosing which reduction technique to choose: the algebraic combination of the objective function, or the neighborhood comparison of the constraints.

In our computational analysis, we compared the AF and the PBC models using benchmark problems of various sizes from the Beasley OR library. Since both models use identical objective functions, the variable reduction was the same. And since the constraints are functionally identical, the constraint reductions were also close to identical. Our main question in comparing the two models was whether the integer-friendly constraints of the PBC model would provide a significant computational advantage over the non-integer friendly constraints of the AF model. The PBC model already has the advantage of flexibility in its constraints; that it may adopt those of either the AlBdaiwi or the Cornuejols model, and likewise for its objective function. For this comparison, we chose to adapt the constraints of the Cornuejols model to the objective function of the AlBdaiwi model in order to provide a pseudo-Boolean formulation of the Cornuejols model.

The results of the computaiton comparison were mildly conclusive. While the PBC model had a slightly faster preprocessing speed and run time, the preprocessing speed is unexpected and unattributed, as the two models have comparable preprocessing algorithms. Additionally, the run time advantage held by the PBC was only moderate, and over a limited sampling of sets. Its consistency, however, is encouraging. For now, we conclude that the PBC model's integer friendly constraint may have a run time advantage over the AF model, and propose further tests to verify it.

When running these computational models, we began to face memory and time issues as the problem sizes increased, which is why our largest problem has only n=500 sites. Different solving software on different computers would likely give different results. Additionally, the code used in preprocessing was less efficient than it could have been: since the two algorithms were near identical, the correctness of the result was more important than the speed of preprocessing.

Additional research might investigate the efficacy of the two reduction methods that were mentioned in this paper but not included in the computational comparison. While their end result is essentially the same, the reduction algorithms might work better in different conditions. If so, it would be useful to determine what these conditions are. Additionally, as suggested by AlBdaiwi et al., a potentially worthwhile course of study would be to investigate the efficacy of the pseudo-Boolean formulation of the p-median problem. Finally, it should remain productive to apply different formulation methods to problems in related fields. As Elloumi did with the simple plant location problem that Cornuejols et al. had originally developed, taking an existing formulation in a related field and adapting it to a current problem can lead to new applications.

Acknowledgements I would especially like to thank Professor Richard Church for the tremendous generosity he has shown with his time. His guidance and insight have been invaluable.

References

[1] AlBdaiwi, B.F., Ghosh, D., and Goldengorin, B. (2011). "Data aggregation for p-median problems." *Journal of Combinatorial Optimization* 21, 348-363.

[2] AlBdaiwi, B.F. and Goldengorin, B. (2009). "Equivalent instances of the simple plant location problem." *Journal of Computation and Applied Mathematics* 57, 812-820.

[3] Avella, P., Sassano, A., and Vasil'ev, I. (2007). "Computational study of large-scale p-median problems." *Mathematical Programming* A 89-114

[4] Boros, E. and Hammer, P.L. (2002). "Pseudo-Boolean optimalization." *Discrete Appiedl Mathematics* 123, 155-225.

[5] Church, R.L. (2003). "COBRA: A new formulation of the classic p-median location problem." *Annals of Operations Research* 122, 103-120.

[6] Cornuejols, G., Nemhauser, G.L., and Wolsey, L.A. (1980). "A canonical representation of simple plant location problems and its applications." *Siam Journal on Matrix Analysis and Applications* 1, 261-272.

[7] Hakimi, S.L. (1964). "Optimum locations of switching centers and the absolute centers and medians of a graph." *Operations Research* 12, 450-459.

[8] Hakimi, S.L. (1965). "Optimum distribution of switching centers in a communication network and some related graph theoretic problems." *Operations Research* 13, 462-475.

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

[10] Revelle, C. S. (1993). "Facility siting and integer-friendly programming." *European Journal of Operational Research* 65, 147-158.

[11] ReVelle, C. S. and Swain, R. W. (1970). "Central facilities location." *Geographical Analysis* 2: 3042.

[12] Rosing, K.E. and Revelle, C.S. (1997). "Heuristic concentration: two-stage solution concentration." *European Journal of Operational Research* 97, 75-86.

[13] Rosing, K.E., Revelle, C.S., and Rosing-Vogelaar, H. (1979). "The p-median and its linear programming relaxation: an approach to large problems." *The Journal of the Operational Research Society* 30, 815-823.

[14] Weber, A. (1909). "Über den Standort der Industrien, Eerster Teil: Reine Theorie des Standortes." Mohr, Tübingen