A Family of Facets for the p-Median Polytope

I. R. de Farias, Jr.
Department of Industrial Engineering
State University of New York at Buffalo
403 Bell Hall, Buffalo, NY 14260-2050
defarias@buffalo.edu

December 9,2000

Abstract

We present a nontrivial family of facet-defining inequalities for the p-median polytope. We incorporate the inequalities in a branch-and-cut scheme, and we report computational results that demonstrate their effectiveness.

Keywords: mixed-integer programming, p-median, facility location, cardinality constrained programming, branch-and-cut

1 Introduction

Let $M = \{1, \ldots, m\}$ be a set of *warehouses*, and $N = \{1, \ldots, n\}$ a set of *clients*. The *p-median problem* (PMP) is to select no more than p warehouses to supply the clients at a minimum cost. Let x_{ij} be the fraction of the demand of client j supplied by warehouse i, $c_{ij} \geq 0$ the cost of sending one unit of flow from warehouse i to client j, and $d_j > 0$ the demand of client j, $i \in M$, $j \in N$. PMP can be formulated as

$$\min \quad \sum_{i \in M} \sum_{j \in N} c_{ij} d_j x_{ij}$$

$$\sum_{i \in M} x_{ij} = 1, \qquad j \in N \tag{1}$$

$$x_{ij} \leq y_i, \qquad i \in M, j \in N \tag{2}$$

$$\sum_{i \in M} y_i \leq p \tag{3}$$

$$x_{ij} \geq 0, \qquad i \in M, j \in N \tag{4}$$

$$y_i \in \{0, 1\}, \qquad i \in M. \tag{5}$$

The *underlying network* of the formulation is the complete bipartite graph $K_{m,n}$. However, this implies in no loss of generality, since PMP in any network can be transformed in polynomial time to PMP in a complete bipartite graph [17]. Many times it is required that *precisely* *p warehouses must be selected*. The results of our study are applicable to this case as well. When $p \geq m$, (3) is redundant, and the feasible set of PMP becomes that of an *uncapacitated facility location problem* (UFL), whose polytope has received considerable attention, see for example [6, 7, 10, 11, 12, 14, 22, 24]. So, we assume that $1 \leq p \leq m-1$.

Besides being interesting and useful in its own right, PMP also appears as a substructure in several other models, see, for instance, [9, 11, 15, 17, 19]. As a consequence, PMP has been extensively studied. In particular, [16] shows that PMP is NP-hard, and [4, 8, 18, 20] present branch-and-bound schemes to solve PMP exactly. However, polyhedral studies for PMP are scarce in the literature. Ward et al. [25] present a polyhedral study for PMP in a tree; Avella and Sassano [3] present two families of facet-defining inequalities valid for the convex hull of the feasible set of PMP in a complete directed graph, and computational results that demonstrate their effectiveness. In this paper we present new polyhedral results for PMP, and we show how they can be used to solve difficult optimization problems that include Constraints (1)-(5).

The organization of the paper is as follows. In Section 2 we discuss the facetial structure of the convex hull of the feasible set of PMP. We establish the trivial families of facet-defining inequalities, and we show that they suffice to describe the convex hull of the feasible set of PMP when $p = 1$. Then, we present a nontrivial family of facet-defining inequalities, and we show that they are not valid for the convex hull of the feasible set of UFL. In Section 3 we present a separation heuristic for the nontrivial family of facet-defining inequalities, and details of our implementation of a branch-and-cut algorithm that uses them as cuts. We describe the instances tested, and we report results of our computational experience with MINTO 3.0 [21, 23], with and without our cuts included, on several instances of PMP with capacity constraints and unsplit demands.

2 The p-Median Polytope

By modifying the objective function coefficients of PMP appropriately, (1) can be replaced by

$$\sum_{i \in M} x_{ij} \leq 1, j \in N, \tag{6}$$

see [13]. Let $S = \{(x, y) : (x, y)$ satisfies (2)-(6)$\}$, PS =conv(S), and $LPS = \{(x, y) : (x, y)$ satisfies (2)-(4), $y \in [0, 1]^m$, and (6)$\}$. It is easy to show that

Proposition 1 *PS is full-dimensional.* □

As a consequence, the inequality representation of the facets of PS is unique up to a multiplicative constant, and thus, from an inequality point of view, it is more convenient to work with the formulation of PMP that contains (6) instead of (1). We call PS the *p-median polytope*, and conv($\{x \in \Re^{mn+m} : x$ satisfies (2), (4), (5), (6)$\}$) the UFL polytope.

The following two propositions are easy to prove.

Proposition 2 *$(x, y) \in \{0,1\}^{mn+m}$ whenever (x, y) is a vertex of PS.* $\square$

Proposition 3 *Inequalities (2), (3), (4), $y_i \leq 1$, and (6) are facet-defining for PS $\forall i \in M, j \in N$.* $\square$

We call the inequalities in Proposition 3 trivial. By repeating the arguments presented in [6] for the UFL polytope, it can be shown that they provide the full inequality description of PS when $m \leq 2$ or $n \leq 2$. Next, we now show that they also provide the full inequality description of PS when $p = 1$. (In the proof of Theorem 1 we use the convention that $\sum_{i=r}^{s} a_i = 0$ when $s < r$.)

Theorem 1 *When $p = 1$, $PS = \{(x, y) \in [0,1]^{mn+m} : x$ satisfies (2), (3), and (6)$\}$.*

Proof We show that no fractional point $(\tilde{x}, \tilde{y}) \in LPS$ can be a vertex of LPS when $p = 1$, by generating a sequence of nonnegative numbers $\lambda_1, \ldots, \lambda_k$ and of points of LPS $(\tilde{x}^1, \tilde{y}^1), \ldots, (\tilde{x}^k, \tilde{y}^k)$, such that

$$\sum_{t=1}^{k} \lambda_t = 1 \tag{7}$$

and

$$\sum_{t=1}^{k} \lambda_t (\tilde{x}^t, \tilde{y}^t) = (\tilde{x}, \tilde{y}). \tag{8}$$

Let $\lambda_1 = \min\{\tilde{x}_{ij} : i \in M, j \in N, \text{ and } \tilde{x}_{ij} > 0\}$. Let $i_1 \in M$ be such that $\tilde{x}_{i_1 j} = \lambda_1$ for some $j \in N$ (in case there are multiple indices $i \in M$ satisfying $\tilde{x}_{ij} = \lambda_1$, choose one arbitrarily), and $(\tilde{x}^1, \tilde{y}^1)$ be given by

$$\tilde{x}^1_{ij} = \begin{cases} 1 & \text{if } i = i_1, j \in N, \text{ and } \tilde{x}_{ij} > 0 \\ 0 & \text{otherwise} \end{cases}$$

and

$$\tilde{y}^1_i = \begin{cases} 1 & \text{if } i = i_1 \\ 0 & \text{otherwise.} \end{cases}$$

For $2 \leq r \leq s-1$, let $\lambda_r = \min\{\tilde{x}_{ij} - \sum_{t=1}^{r-1} \lambda_t \tilde{x}^t_{ij} : i \in M, j \in N, \text{ and } \tilde{x}_{ij} - \sum_{t=1}^{r-1} \lambda_t \tilde{x}^t_{ij} > 0\}$, where s is such that $\tilde{x}_{ij} - \sum_{t=1}^{s-2} \lambda_t \tilde{x}^t_{ij} > 0$ for some $i \in M, j \in N$, and $\tilde{x}_{ij} - \sum_{t=1}^{s-1} \lambda_t \tilde{x}^t_{ij} = 0$ $\forall i \in M, j \in N$. Let $i_r \in M$ be such that $\tilde{x}_{i_r j} - \sum_{t=1}^{r-1} \lambda_t \tilde{x}^t_{i_r j} = \lambda_r$ for some $j \in N$ (in case there are multiple indices $i \in M$ satisfying $\tilde{x}_{ij} - \sum_{t=1}^{r-1} \lambda_t \tilde{x}^t_{ij} = \lambda_r$, choose one arbitrarily), and $(\tilde{x}^r, \tilde{y}^r)$ be given by

$$\tilde{x}^r_{ij} = \begin{cases} 1 & \text{if } i = i_r, j \in N, \text{ and } \tilde{x}_{ij} - \sum_{t=1}^{r-1} \lambda_t \tilde{x}^t_{ij} > 0 \\ 0 & \text{otherwise.} \end{cases}$$

and

$$\tilde{y}^r_i = \begin{cases} 1 & \text{if } i = i_r \\ 0 & \text{otherwise.} \end{cases}$$

For $s \le v \le k-1$, let $\lambda_v = \min\{\tilde{y}_i - \sum_{t=1}^{v-1} \lambda_t \tilde{y}^t_i : i \in M \text{ and } \tilde{y}_i - \sum_{t=1}^{v-1} \lambda_t \tilde{y}^t_i > 0\}$, where k is such that $\tilde{y}_i - \sum_{t=1}^{k-2} \lambda_t \tilde{y}^t_i > 0$ for some $i \in M$, and $\tilde{y}_i - \sum_{t=1}^{k-1} \lambda_t \tilde{y}^t_i = 0 \ \forall i \in M$. Let $M_v = \{i \in M : \tilde{y}_i - \sum_{t=1}^{v-1} \lambda_t \tilde{y}^t_{ij} = \lambda_v\}$, and $(\tilde{x}^v, \tilde{y}^v)$ be given by $\tilde{x}^v = 0$ and

$$\tilde{y}^v_i = \begin{cases} 1 & \text{if } i \in M_v \\ 0 & \text{otherwise.} \end{cases}$$

Finally, let $\lambda_k = 1 - \sum_{t=1}^{k-1} \lambda_t$, and $(\tilde{x}^k, \tilde{y}^k) = (0, 0)$. It follows that $\sum_{t=1}^{k-1} \lambda_t = \sum_{i=1}^{m} \tilde{y}_i$. Because $p = 1$, (3) implies that $\lambda_k \ge 0$, and that (7) holds. Also, by construction, (8) holds. □

We now present a family of facet-defining inequalities for *PS* that are not valid for the UFL polytope.

Theorem 2 *Let $N' \subseteq N$ be such that $|N'| \ge p+1$, and M_j, $j \in N'$, be nonempty disjoint subsets of M. Let $M' = M - \cup_{j \in N'} M_j$, and suppose that $M' \ne \emptyset$. Then,*

$$\sum_{j \in N'} \sum_{i \in M_j} x_{ij} + \sum_{j \in N'} \sum_{i \in M'} x_{ij} \le p + (|N'| - p) \sum_{i \in M'} y_i \tag{9}$$

is valid and facet-defining for PS. □

Before proving Theorem 2, we illustrate it with an example.

Example 1 Let $m = 4$, $n = 3$, and $p = 2$. Then,

$$x_{11} + x_{22} + x_{33} + x_{41} + x_{42} + x_{43} \le 2 + y_4 \tag{10}$$

defines a facet of *PS* with $N' = N$, $M_1 = \{1\}$, $M_2 = \{2\}$, $M_3 = \{3\}$, and $M' = \{4\}$. Inequality (10) cuts off the point $(\tilde{x}, \tilde{y})$ given by $\tilde{x}_{11} = \tilde{y}_1 = \tilde{x}_{22} = \tilde{y}_2 = \tilde{x}_{33} = \tilde{y}_3 = \tilde{x}_{41} = \tilde{x}_{42} = \tilde{x}_{43} = \tilde{y}_4 = \frac{1}{2}$, and $\tilde{x}_{12} = \tilde{x}_{13} = \tilde{x}_{21} = \tilde{x}_{23} = \tilde{x}_{31} = \tilde{x}_{32} = 0$, which is a vertex of *LPS*. Note that $(\tilde{x}, \tilde{y})$ belongs to the UFL polytope. This shows that (9) is not valid for the UFL polytope. □

Proof of Theorem 2 We first prove that (9) is valid. Let $(\tilde{x}, \tilde{y})$ be a vertex of PS. It follows that $(\tilde{x}, \tilde{y}) \in \{0,1\}^{mn+m}$. Suppose that $\tilde{y}_i = 0 \ \forall i \in M'$. Then,

$$p + (|N'| - p) \sum_{i \in M'} \tilde{y}_i = p.$$

Because of (2), $\tilde{x}_{ij} = 0 \ \forall i \in M', j \in N'$. For $i \in M_j$, the only variable with index i present in (9) is x_{ij}. Since at most p variables with distinct warehouse indices can be positive in a vertex of PS,

$$\sum_{j \in N'} \sum_{i \in M_j} \tilde{x}_{ij} + \sum_{j \in N'} \sum_{i \in M'} \tilde{x}_{ij} = \sum_{j \in N'} \sum_{i \in M_j} \tilde{x}_{ij} \leq p.$$

If $\tilde{y}_i = 1$ for some $i \in M'$, it follows that

$$p + (|N'| - p) \sum_{i \in M'} \tilde{y}_i \geq |N'|.$$

On the other hand,

$$\sum_{j \in N'} \sum_{i \in M_j} \tilde{x}_{ij} + \sum_{j \in N'} \sum_{i \in M'} \tilde{x}_{ij} \leq \sum_{j \in N'} \sum_{i \in M} \tilde{x}_{ij} \leq |N'|,$$

where the last inequality follows from (6).

We now give $mn + m$ linearly independent points of PS that satisfy (9) at equality. Let $x^{(r,s)} \in \Re^{mn}$, $r \in M$, $s \in N$, and $y^{(r)} \in \Re^m$, $r \in M$, be such that

$$x_{ij}^{(r,s)} = \begin{cases} 1 & \text{if } (i,j) = (r,s) \\ 0 & \text{otherwise} \end{cases}$$

and

$$y_i^{(r)} = \begin{cases} 1 & \text{if } i = r \\ 0 & \text{otherwise.} \end{cases}$$

The points $P^1(r) = (\sum_{j \in N'} x^{(r,j)}, y^{(r)})$, $r \in M'$, belong to PS and satisfy (9) at equality. Let $t \in M'$. The points $P^2(r) = P^1(t) + (0, y^{(r)}), r \in \cup_{j \in N'} M_j$, belong to PS and satisfy (9) at equality. The same is also true for $P^3(r,s) = P^2(r) + (x^{(r,s)}, 0), r \in \cup_{j \in N'} M_j, s \in N - N'$, and $P^4(r,s) = P^1(r) + (x^{(r,s)}, 0), r \in M', s \in N - N'$. Because $|N'| \geq p + 1$, PS has $(\sum_{j \in N'} |M_j|) \times |N'|$ linearly independent points, $P^5(r,s), r \in \cup_{j \in N'} M_j$, $s \in N'$, that satisfy (9) at equality, with $x_{ij} = 0 \ \forall (i,j) \in (M \times (N - N')) \cup (M' \times N')$, $y_i = 0 \ \forall i \in M'$, and the square matrix B formed with the components $x_{ij}, i \in \cup_{j \in N'} M_j, j \in N'$, is nonsingular. Finally, let $u \in \cup_{j \in N'} M_j$. The points $P^6(r,s) = P^1(r) + (x^{(u,s)} - x^{(r,s)}, y^{(u)})$, $r \in M'$ and $s \in N'$, belong to PS and satisfy (9) at equality.

Consider now the block matrix

$$\begin{pmatrix} 0 & A & 0 & 0 & 0 & I \\ 0 & 0 & 0 & 0 & I & 0 \\ 0 & 0 & I & 0 & 0 & 0 \\ 0 & 0 & 0 & I & 0 & 0 \\ B & 0 & 0 & 0 & C & 0 \\ D & -I & 0 & 0 & E & 0 \end{pmatrix}$$

with the points $P^1(r)$ in the first rows, followed by the points $P^2(r) - P^1(t)$, followed by $P^3(r,s) - P^2(r)$, then by $P^4(r,s) - P^1(r)$, $P^5(r,s)$, and finally $P^6(r,s) - P^1(r)$. The first block of columns correspond to the variables x_{ij} with $i \in \cup_{j\in N'} M_j$ and $j \in N'$. The second block of columns correspond to the variables x_{ij} with $i \in M'$ and $j \in N'$. The third block of columns correspond to the variables x_{ij} with $i \in \cup_{j\in N'} M_j$ and $j \in N - N'$. The forth block of columns correspond to the variables x_{ij} with $i \in M'$ and $j \in N - N'$. The fifth block of columns correspond to the variables y_i with $i \in \cup_{j\in N'} M_j$, and the last block of columns correspond to the variables y_i with $i \in M'$. The identity matrices are obtained possibly after reordering some of the columns. Clearly the matrix is nonsingular, and therefore, (9) is facet-defining. □

3 Computational Experience

Although PMP is NP-hard, in general it is easy to solve it exactly. As a matter of fact, it is not uncommon to obtain an optimal solution to PMP simply by solving the LP relaxation. As part of our preliminary computational experiments, we tested the data available in [5] with CPLEX 6.5. CPLEX was able to solve to proven optimality all instances, except the ones with $m, n \in \{800, 900\}$. The instances solved to proven optimality required on average 12 nodes, none required over 30, most required only 5, and several were solved at the root node. Because the time needed to solve the LP relaxations for the instances with $m, n \in \{800, 900\}$ was too great, we stopped CPLEX after 6 nodes in these cases. Even then, the final integrality gap was less than 1%.

On the other hand, solving practical instances of PMP with capacity constraints is usually difficult. Our objective is to test the effectiveness of (9) within a branch-and-cut scheme to solve hard optimization problems that include Constraints (1)-(5). We then implemented (9) in MINTO 3.0 [21, 23], and we tested it on several instances of PMP with capacity constraints and unsplit demands (all flow to a particular customer must come from the same warehouse.) The capacity constraints are

$$\sum_{j\in N} d_j x_{ij} \leq s_i, i \in M,$$

where s_i is the total supply of warehouse i. Unsplit demands are enforced by requiring $x_{ij} \in \{0, 1\}$.

We tested instances with $m \in \{30, 40, 50, 60, 70\}$ and, for each m, $n \in \{m, m+10, m+20, \ldots, 100\}$, 3 instances for each pair (m, n). We performed our computational experiments on a Sun Ultra 2 with two UltraSPARC 300 MHz CPUs and 256 MB memory. We limited the total CPU time for each test to 100,000 seconds.

To generate an instance with m warehouses and n clients, we first generated n points uniformly in a unit square. All points correspond to client sites, and the first m points correspond also to warehouse sites. c_{ij} is defined to be the euclidean distance between warehouse i and client j. Note that $c_{ii} = 0\ \forall i \in M$. The demands were uniformly generated from the interval [5,35], and the supplies from [55,300], with the exception of warehouse m, whose supply is $\sum_{j \in N} d_j$, in order to avoid infeasibility. In our preliminary computational experiments, the instances with small values of p appeared to be the hardest. So we chose $p = 3$ for $m \in \{30, 40, 50\}$ and $p = 5$ for $m \in \{60, 70\}$.

3.1 Branch-and-Cut Alternatives

We performed preliminary tests with MINTO's pre-processing and node selection alternatives. The ones that performed best were: do pre-processing and limited probing, use best bound for node selection, and use a pseudo-cost variable rule for variable branching selection. We then used these options in our computations. We also tested different alternatives with respect to row management, i.e. perform no row management, delete constraints with positive slacks every 10 iterations, etc. Besides (9), MINTO generated two families of cuts automatically, clique inequalities (CIs) at the root node, and lifted cover inequalities (LCIs) at all nodes (see [21, 23] for a comprehensive description of the cuts available in MINTO). The row management alternative that performed best in our preliminary tests was to delete all LCIs and (9) with positive slacks every 100 nodes, and we adopted this alternative.

3.2 Separation Heuristic

Given a point $(\tilde{x}, \tilde{y}) \in LPS$, to find a violated Inequality (9), one has to select the subsets $N' \subseteq N$ with $|N'| \geq p+1$, and $M_j \subset M$, $j \in N'$, such that $M' = M - \cup_{j \in N'} M_j$ is nonempty, and

$$\sum_{j \in N'} \sum_{i \in M_j} \tilde{x}_{ij} + \sum_{j \in N'} \sum_{i \in M'} \tilde{x}_{ij} > p + (|N'| - p) \sum_{i \in M'} \tilde{y}_i. \tag{11}$$

It appears then that the separation problem for (9) is difficult. Thus, we used a heuristic, which we now describe, to solve the separation problem.

Because $|M'|$ must be positive, we first select an element $i' \in M'$. We select i' in a greedy fashion to be such that $\sum_{j \in N} \tilde{x}_{i'j}$ =max$\{\sum_{j \in N} \tilde{x}_{ij} : i \in M\}$, hoping that $\sum_{j \in N'} \sum_{i \in M'} \tilde{x}_{ij}$ is sufficiently great at the end. Then we select N' and $M_j, j \in N'$, simultaneously, and in a greedy fashion too. For each $i \in M - \{i'\}$ for which $\sum_{k \in N} \tilde{x}_{ik} > 0$, we determine $j \in N$ such that $\tilde{x}_{ij}$ =max$\{\tilde{x}_{ik} : k \in N\}$, and we include j in N' and i in M_j, i.e. $N' \leftarrow N' \cup \{j\}$ and $M_j \leftarrow M_j \cup \{i\}$. At the end, $M' \leftarrow M' \cup (M - \cup_{j \in N'} M_j)$. If $|N'| \geq p+1$, and (11) holds, we include the corresponding Inequality (9) in the constraint set of PMP at the node under consideration, and we solve the new LP relaxation. Otherwise, we branch.

Example 2 Consider the instance and the point $(\tilde{x}, \tilde{y})$ defined in Example 1. Since $\sum_{j=1}^{3} \tilde{x}_{4j} = \frac{3}{2}$, while $\sum_{j=1}^{3} \tilde{x}_{1j} = \sum_{j=1}^{3} \tilde{x}_{2j} = \sum_{j=1}^{3} \tilde{x}_{3j} = \frac{1}{2}$, $i' = 4$, and $M' \leftarrow \{4\}$. Since $\sum_{j=1}^{3} \tilde{x}_{1j}$, $\sum_{j=1}^{3} \tilde{x}_{2j}$, $\sum_{j=1}^{3} \tilde{x}_{3j}$ are positive, and $\tilde{x}_{11} = \max\{\tilde{x}_{11}, \tilde{x}_{12}, \tilde{x}_{13}\}$, $\tilde{x}_{22} = \max\{\tilde{x}_{21}, \tilde{x}_{22}, \tilde{x}_{23}\}$, and $\tilde{x}_{33} = \max\{\tilde{x}_{31}, \tilde{x}_{32}, \tilde{x}_{33}\}$, $M_1 = \{1\}$, $M_2 = \{2\}$, $M_3 = \{3\}$, and $M' = \{4\}$. The resulting inequality is (10), which cuts off $(\tilde{x}, \tilde{y})$. $\square$

3.3 Computational Results

Table 1 gives, for each instance size, the average number of nodes, computational time, and the number of cuts generated, over the 3 instances tested, with and without (9). The instance sizes are denoted as $m.n$. For the instances MINTO was unable to complete enumeration within 100,000 seconds, the total CPU time given for that instance was 100,000 seconds, and the number of nodes was the number of nodes that had been generated so far.

Preliminary tests showed that LCIs and CIs are instrumental in closing the integrality gap. However, MINTO without (9) was unable to end with a proven optimal solution for 43% of the instances tested. On the other hand, with (9) included, MINTO was able to end with a proven optimal solution for all 90 instances. Note that the number of CIs and LCIs generated is greatly reduced when (9) is included. In all instances, the inclusion of (9) was very effective in reducing the number of nodes. However, for the smaller instances, the inclusion of cuts was prohibitively time consuming. A possible way around it is to not introduce (9) in all nodes, or maybe to limit the total number of Inequalities (9) generated.

Overall, the average reduction in number of nodes was 92% and in computational time 71%. The results clearly indicate that (9) can be helpful in solving exactly difficult problems that include Constraints (1)-(5), and that the polyhedral investigation of *PS* can be of practical use.

Table 1: Number of nodes, computational time, and number of cuts

Inst. size	Nodes W/O (9)	Nodes W/ (9)	Nodes % Red.	Time W/O (9)	Time W/ (9)	Time % Red.	Cuts W/O (9) LCI	Cuts W/O (9) CI	Cuts W/ (9) LCI	Cuts W/ (9) CI	Cuts W/ (9) (9)
30.30	333	149	55	58	73	-20	27	45	10	18	129
30.40	222	123	44	75	184	-59	1	5	0	16	115
30.50	375	154	58	302	721	-58	10	19	0	35	135
30.60	670	350	47	734	584	20	187	280	6	111	226
30.70	314	127	59	1,323	1,034	21	259	509	92	165	105
30.80	874	243	72	3,944	2,811	28	78	162	81	178	123
30.90	2,953	1,798	39	1,036	760	26	231	612	26	118	664
30.100	28,545	2,042	92	52,709	14,021	73	290	547	76	130	588
40.40	81	44	45	38	26	31	19	37	1	31	2
40.50	3,525	1,129	67	44,282	24,213	45	923	1,905	547	1,249	622
40.60	1,077	585	45	5,432	6,704	-18	238	396	139	451	316
40.70	3,995	2,601	34	5,483	5,823	-5	197	268	12	53	585
40.80	1,787	999	44	10,835	13,565	-20	120	288	101	577	315
40.90	29,237	651	97	100,000	14,757	85	257	929	175	427	305
40.100	27,891	3,814	86	100,000	46,378	53	322	876	152	646	1,352
50.50	226	76	66	493	545	-9	12	42	1	15	104
50.60	1,663	1,186	28	844	1,060	-20	12	38	9	76	21
50.70	5,841	736	87	22,818	5,866	74	1,271	3,259	221	542	318
50.80	26,332	267	98	100,000	9,683	90	325	722	79	170	200
50.90	24,282	552	97	100,000	21,327	78	373	640	144	169	311
50.100	23,717	300	98	100,000	14,239	85	126	630	237	589	126
60.60	2,298	1,097	52	2,067	960	53	63	211	31	108	71
60.70	28,515	1,678	94	100,000	41,298	58	287	569	231	782	1,033
60.80	19,458	100	99	100,000	8,173	91	367	813	250	921	3,102
60.90	16,722	1,742	89	100,000	60,124	39	312	984	430	1,053	900
60.100	16,860	465	97	100,000	20,760	79	147	450	380	781	2,153
70.70	14,525	412	97	100,000	29,061	70	318	634	131	412	325
70.80	17,589	412	97	100,000	35,478	64	235	671	1,197	425	386
70.90	15,327	84	99	100,000	16,144	83	412	687	27	91	175
70.100	12,831	100	99	100,000	23,759	76	757	938	197	341	238
Total	328,065	24,016	92	1,452,473	420,176	71	8,176	18,166	4,983	10,680	15,045

References

[1] K. Aardal, "Capacitated Facility Location: Separation Algorithms and Computational Experience," *Mathematical Programming* 81, 149-175 (1998).

[2] K. Aardal, I. Pochet, and L.A. Wolsey, "Capacitated Facility Location: Valid Inequalities and Facets," *Mathematics of Operations Research* 20, 562-582 (1995).

[3] P. Avella and A. Sassano, "On the p-Median Polytope," to appear in Mathematical Programming.

[4] J.E. Beasley, "A Note on Solving Large p-Median Problems," *European Journal of Operational Research* 21, 270-273 (1985).

[5] J.E. Beasley, "OR-Library," http://mscmga.ms.ic.ac.uk/info.html.

[6] D.C. Cho, E.L. Johnson, M.W. Padberg, and M.R. Rao, "On the Uncapacitated Location Problem I: Valid Inequalities and Facets," *Mathematics of Operations Research* 8, 579-589 (1983).

[7] D.C. Cho, M.W. Padberg, and M.R. Rao, "On the Uncapacitated Location Problem II: Facets and Lifting Theorems," *Mathematics of Operations Research* 8, 590-612 (1983).

[8] N. Christofides and J.E. Beasley, "A Tree Search Algorithm for the p-Median Problem," *European Journal of Operational Research* 10, 196-204 (1982).

[9] G. Cornuejols, M.L. Fisher, and G.L. Nemhauser, "Location of Bank Accounts to Optimize Float: An Analytic Study of Exact and Approximate Algorithms," *Management Science* 23, 789-810 (1977).

[10] G. Cornuejols, M.L. Fisher, and G.L. Nemhauser, "On the Uncapacitated Location Problem," *Annals of Discrete Mathematics* 1, 163-177 (1977).

[11] G. Cornuejols, G.L. Nemhauser, and L.A. Wolsey, "The Uncapacitated Facility Location Problem," in P.B. Mirchandani and R.L. Francis, eds., *Discrete Location Theory*, 1-54, John Wiley and Sons (1990).

[12] G. Cornuejols and J.M. Thizy, "Some Facets of the Simple Plant Location Polytope," *Mathematical Programming* 23, 50-74 (1982).

[13] I.R. de Farias, "A Polyhedral Approach to Combinatorial Complementarity Programming Problems," *Ph.D. Thesis*, School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, GA (1995).

[14] M. Guignard, "Fractional Vertices, Cuts and Facets of the Simple Plant Location Problem," *Mathematical Programming Study* 12, 150-162 (1980).

[15] S.L. Hakimi and E.F. Schmeichel, "Locating Replicas of a Database on a Network," *Networks* 30, 31-36 (1997).

[16] O. Kariv and S.L. Hakimi, "An Algorithmic Approach to Network Location Problems II: the p-Medians," *SIAM Journal on Applied Mathematics* 37, 539-560 (1979).

[17] J. Krarup and P.M. Pruznan, "Ingredients of Locational Analysis," in P.B. Mirchandani and R.L. Francis, eds., *Discrete Location Theory*, 1-54, John Wiley and Sons (1990).

[18] P. Järvinen, J. Rajala, and H. Sinervo, "A Branch-and-Bound Algorithm for Seeking the p-Median," *Operations Research* 20, 173-178 (1972).

[19] P.B. Mirchandani "The p-Median Problem and Generalizations," in P.B. Mirchandani and R.L. Francis, eds., *Discrete Location Theory*, 1-54, John Wiley and Sons (1990).

[20] S.C. Narula, U.I. Ogbu, and H.M. Samuelsson, "An algorithm for the p-Median Problem," *Operations Research* 25, 709-713 (1977).

[21] G.L. Nemhauser, G.C. Sigismondi, and M.W.P. Savelsbergh, "MINTO, a Mixed-INTeger Optimizer," *Operations Research Letters* 15, 47-58 (1994).

[22] M.W. Padberg, T.J. van Roy, and L.A. Wolsey, "Valid Linear Inequalities for Fixed Charge Problems," *Operations Research* 33, 842-861 (1985).

[23] Savelsbergh, M.W.P., "Functional Description of MINTO, a Mixed INTeger Optimizer (version 3.0)," http://udaloy.isye.gatech.edu/ mwps/projects/minto.html.

[24] T.J. van Roy and L.A. Wolsey, "Valid Inequalities and Separation for Uncapacitated Fixed Charge Networks," *Operations Research Letters* 4, 105-112 (1985).

[25] R.T. Ward, P. Lemke, and A. Oudjit, "Properties of the k-Median Linear Programming Relaxation," Technical Report, Rensselaer Polytechnic Institute, Troy, NY (1988).