Journal of Algorithms **31,** 228–248 (1999)
Article ID jagm.1998.0993, available online at http://www.idealibrary.com on IDEAL®

Greedy Strikes Back: Improved Facility Location Algorithms*

Sudipto Guha†

Department of Computer Science, Stanford University, Stanford, California 94305

and

Samir Khuller‡

Computer Science Department and UMIACS, University of Maryland, College Park, Maryland 20742

Received November 22, 1997; revised September 29, 1998

A fundamental facility location problem is to choose the location of facilities, such as industrial plants and warehouses, to minimize the cost of satisfying the demand for some commodity. There are associated costs for locating the facilities, as well as transportation costs for distributing the commodities. We assume that the transportation costs form a metric. This problem is commonly referred to as the *uncapacitated facility location* problem. Application to bank account location and clustering, as well as many related pieces of work, are discussed by Cornuejols, Nemhauser, and Wolsey. Recently, the first constant factor approximation algorithm for this problem was obtained by Shmoys, Tardos, and Aardal. We show that a simple greedy heuristic combined with the algorithm by Shmoys, Tardos, and Aardal, can be used to obtain an approximation guarantee of 2.408. We discuss a few variants of the problem, demonstrating better approximation factors for restricted versions of the problem. We also show that the problem is max SNP-hard. However, the inapproximability constants derived from the max SNP hardness are very close to one. By relating this problem to Set Cover, we prove a lower bound of 1.463 on the best possible approximation ratio, assuming $NP \notin DTIME[n^{O(\log\log n)}]$.

* A preliminary version of this paper appeared in the *Proceedings of the Ninth Annual ACM-SIAM Symposium on Discrete Algorithms (1998)*.

† Supported by an ARO MURI Grant DAAH04-96-1-0007 and NSF Award CCR-9357849, with matching funds from IBM, Schlumberger Foundation, Shell Foundation, and Xerox Corporation. E-mail: sudipto@cs.stanford.edu.

‡ Research supported by NSF CAREER Award CCR-9501355. E-mail: samir@cs.umd.edu.

0196-6774/99 $30.00

1. INTRODUCTION

We consider the *uncapacitated facility location problem* (UFL), which is defined as follows. We are given a set of demand locations D where clients require service, and a set of locations F such that we may build a facility at location i with cost f_i. If a client at location j with demand d_j is assigned to a facility at location i we incur a service cost of $d_j \cdot c_{ij}$. Our objective is to minimize the *total* cost of building facilities and servicing the clients. We assume that the service costs c_{ij} form a metric. Not surprisingly, this problem is NP-hard.

An approximation algorithm with a factor of ρ is a polynomial time algorithm that produces a solution whose total cost is at most ρ times the optimal solution cost. Hochbaum [6] showed that this can be approximated with a factor of $O(\log n)$ by using a greedy algorithm. Very recently, Shmoys, Tardos, and Aardal [16] gave the first constant factor approximation algorithm that achieves an approximation ratio of 3.16. This algorithm is based on a method introduced by Lin and Vitter [10] for rounding fractional solution to linear programming relaxations. Recently and independently, Korupolu, Plaxton, and Rajaraman [8] have shown that the local search heuristic proposed by Kuehn and Hamburger [9] achieves a constant approximation for many variants of the UFL problem.

Uncapacitated facility location (UFL) problems have been *widely studied* in the *operation research* literature (see [1, 9, 12, 17]). This problem has been studied from different perspectives, as clustering analysis [14], communication networks [13], and many others. For an extensive survey of closely related work one can read the chapter by Cornuejols, Nemhauser, and Wolsey [2].

Outline of Results

It is easy to show that the problem is max SNP-hard (Subsection 3.1). We prove that if there is a polynomial time algorithm that approximates the problem to a factor better that 1.463 then $NP \subseteq DTIME[n^{O(\log \log n)}]$. This hardness result is established through hardness proofs for the set cover problem [11, 4] (Subsection 3.2).

We also show that a simple and natural greedy "local-improvement" algorithm, combined with the algorithm given by Shmoys, Tardos, and Aardal [16], yields an approximation factor of 2.408. This algorithm and its proof are described in Section 4. In Section 5 we present a simple family of examples for which the "gap" between the integral and fractional solution for a natural linear programming (LP) formulation studied by Shmoys, Tardos, and Aardal [16] is a factor of 1.463.

The hardness results are proven, assuming that the facility costs for locations in F are identical. In this special case when facility costs are uniform, we can improve the approximation ratio to 2.225 (Subsection 4.4).

In Section 6 we consider a model where every location is a facility and demand vertex. We refer to this problem as the *complete metric uncapacitated facility location problem*. We use the same techniques to show that this simplified version when facilities can be built anywhere, and all facilities cost the same—has an inapproximability bound of 1.278 unless $NP \subseteq DTIME[n^{O(\log\log n)}]$. For this model, where facilities can be built at any location at unit cost, we give an algorithm which achieves an approximation guarantee of 2.104. Thus we demonstrate varying degrees of hardness and possible approximation factors for several variations of the basic problem.

Very recently, Chudak [3] has shown that the approximation factor can be improved by using randomized rounding and the optimal dual LP solution.

2. PRELIMINARIES

In this section we discuss the basic notation used in the Shmoys, Tardos, and Aardal [16] paper. We are given a set of location $N = \{1, \dots, n\}$, a subset $F \subseteq N$ of locations at which we may open a facility, and a subset $D \subseteq N$ of locations that must be assigned to an open facility. For each location $j \in D$ there is a positive demand d_j that must be shipped from its assigned location. For each location $i \in F$ the cost of opening a facility is f_i. The service cost of assigning location $j \in D$ to an open facility at i is c_{ij} per unit of demand shipped. We assume that the c_{ij} $(i, j \in N)$ costs are nonnegative, symmetric, and satisfy the triangle inequality. We wish to find an assignment of each location in D to an open facility so as to minimize the total cost incurred. This is the *uncapacitated facility location problem* (UFL).

We can state the above problem as an integer programming problem as in [1]. There is a 0/1 variable y_i, $i \in F$, indicating if a facility is open at location i. The 0/1 variable x_{ij}, $i \in F$, $j \in D$, indicates if client j is assigned to the facility at i:

$$\min \sum_{i \in F} f_i y_i + \sum_{i \in F} \sum_{j \in D} d_j c_{ij} x_{ij}$$

subject to

$$\begin{aligned} \sum_{i \in F} x_{ij} &= 1 && \text{for each } j \in D \\ x_{ij} &\leq y_i && \text{for each } i \in F, j \in D \\ x_{ij} &\in \{0, 1\} && \text{for each } i \in F, j \in D \\ y_i &\in \{0, 1\} && \text{for each } i \in F. \end{aligned}$$

Shmoys, Tardos, and Aardal [16] derive an algorithm that is based on solving a linear relaxation of the above integer program by removing the integrality requirements for the variables and simply requiring that $x_{ij} \geq 0$ and $y_i \geq 0$. We solve this LP and then the obtained optimal (fractional) solution is rounded to an integer solution.

Let us define λ_f to be the cost of opening facilities in the optimal integral solution, and λ_c to be the service cost corresponding to this optimal integral solution.

Our linear programming formulation is the same as in [1, 16], with an extra constraint that bounds the cost of opening facilities. In other words, we add the new constraint that

$$\sum_{i \in F} f_i y_i \leq \lambda_f$$

for a given value λ_f which it the "budget" for the cost of constructing facilities. Once we do this, we can eliminate the facility construction cost from the objective function and simply minimize the service cost. The reason for doing this will become clear when we discuss the algorithm. Since we do not "know" λ_f, we will have to try $\lceil \log_{1+\epsilon} \mathscr{F} \rceil$ values to guess it within an $(1 + \epsilon)$ factor, where $\mathscr{F} = \sum_{i \in F} f_i$ and for all i, $f_i \geq 1$. Notice that after adding this constraint, we still have a fractional solution whose service cost is at most λ_c and facility construction cost is at most λ_f, since there is an integral solution (the optimal integral solution) that satisfies these requirements.

3. HARDNESS RESULTS

3.1. *Max SNP Hardness*

The following problem was shown to be max SNP-hard by Papadimitriou and Yannakakis [15].

B-vertex cover. Given a graph $G = (V, E)$, with the degree of each node bounded by a constant B, find the minimum cardinality vertex cover. A vertex cover is a subset of vertices such that each edge in G has at least one end point in the cover.

We prove the following theorem via an L-reduction from the B-vertex cover.

THEOREM 3.1. *The uncapacitated facility location problem is max SNP-hard.*

Proof. The B-vertex cover problem has been shown to be max SNP-hard by Papadimitriou and Yannakakis [15]. We show that for any $\epsilon > 0$, there exists an $\epsilon' > 0$ such that an $(1 + \epsilon')$-approximation algorithm for UFL implies a $(1 + \epsilon)$-approximation for the B-vertex cover. Given an instance of the B-vertex cover, we construct an instance of the UFL problem, where the vertices of G correspond to facility locations, and the edges correspond to demand locations. The distance (service cost) between facility i (representing vertex i) and demand location j (representing edge j) is 1 if edge j is incident to vertex i, and 2 otherwise. Let the optimal vertex cover have size k (we may assume that we know this, as $1 \le k \le |V|$, and we can try each choice for k). The cost of opening a facility is $|E|/Bk$. All demands are 1.

Suppose that the UFL problem can be approximated within a factor of $1 + \epsilon'$, $(\epsilon' = \epsilon/(1 + B))$. Corresponding to the optimal vertex cover solution, there exists an assignment of cost at most $(|E| + |E|/B)$. We are guaranteed a solution of cost at most $(|E| + |E|/B)(1 + \epsilon/(1 + B)) = |E| + (1 + \epsilon)(|E|/B)$.

Suppose the algorithm chooses βk facilities and covers $t|E|$ edges at distance exactly 2. The total cost is $\beta(|E|/B) + |E| + t|E|$. This cost can be bounded by $|E| + (1 + \epsilon)(|E|/B)$; thus $\beta + tB \le (1 + \epsilon)$.

By choosing the vertices corresponding to the open facilities and one vertex for each of $t|E|$ edges, we obtain a vertex cover. Since $Bk \ge |E|$, the size of the vertex cover is $\beta k + t|E| \le (\beta + tB)k \le (1 + \epsilon)k$. ∎

Since the best known lower bound for the vertex cover problem is $\frac{7}{6}$ [5], this does not give very strong hardness bounds. We present a scheme to produce much stronger lower bounds, based on the recent results on the hardness of approximating the set cover problem [4].

3.2. *Improved Hardness Results*

We establish a relationship between the set cover problem and the UFL problem and show that if there is a polynomial time algorithm for the UFL problem with an approximation factor smaller than 1.463 then we have a polynomial time approximation algorithm for set cover that has an approximation ratio of $c \ln|X|$ for some constant $c < 1$. By Feige's result [4], this implies that $NP \subseteq DTIME[n^{O(\log\log n)}]$.[1]

Consider an instance of set cover defined as follows: let the set of elements be $X = \{x_1, \ldots, x_n\}$. Let the collection of sets be defined as $\mathscr{S} = \{S_1, S_2, \ldots, S_m\}$, where each $S_i \subseteq X$. Moreover, we may assume that k is the size of the minimum set cover and is known to the approximation algorithm. (This is not a restricting assumption, since $1 \leq k \leq m$, and we can run a set cover algorithm for each possible value of k.)

THEOREM 3.2. *If there is a polynomial time algorithm with an approximation factor smaller than* 1.463 *for the metric uncapacitated facility location problem then* $NP \subseteq DTIME[n^{O(\log\log n)}]$.

Proof. We transform an instance of set cover (X, S) to an instance of the UFL problem. We set $D = X$ and $F = S$. Each f_i value (facility construction cost) is identical. If $x_j \in S_i$ then $c_{ij} = 1$ (in other words, distances are 1 to the sets (facilities) that this element (demand point) belongs to). We extend c_{ij} to be a metric using shortest paths. We set $d_j = 1$ for each demand point j.

The basic idea is to use the approximation algorithm for solving UFL to find partial solutions for the set cover instance, by repeatedly running the UFL algorithm. If the algorithm repeatedly finds a small partial cover, covering a large fraction of the elements, then we obtain a $c \ln|X|$ approximation for the set cover, with $c < 1$. Using Feige's result [4] we get that $NP \subseteq DTIME[n^{O(\log\log n)}]$. Otherwise, if at any iteration the algorithm does not cover a large fraction of the elements, we obtain a lower bound on the approximation ratio.

We now describe the proof more formally. Consider the algorithm SET COVER $(X, \mathscr{S})$ to solve an instance of the set cover problem. Let γ be a constant whose value will be specified later. We assume that UFL-Approx Algorithm (F, D) is a polynomial time α approximation algorithm for UFL.

SET COVER $(X, \mathscr{S})$.

1. Create a UFL instance (F, D) corresponding to $(X, \mathscr{S})$
2. Let $f = \gamma(|D|/k)$ (cost of a facility)
3. **while** $D \neq \emptyset$ **do**
4. $\quad F' =$ UFL-Approx-Algorithm (F, D)
5. $\quad$ Let D' be the clients covered at distance 1

[1] $DTIME[f(n)]$ is the class of decision problems that can be solved in (deterministic) time $O(f(n))$.

6. Let $F = F - F'$ and $D = D - D'$
7. Let $f = \gamma(|D|/k)$ be the cost of a facility
8. **end-proc**

Suppose that in iteration j, $n_j = |D|$ and $f^{(j)}$ is the cost of a facility (uniform costs for all facilities). Since there is a solution of size k to the set cover problem that covers all the elements, there is a solution to the facility location problem of cost $f^{(j)}k + n_j$ (we build facilities at the k locations corresponding to the set cover; all the service costs are 1; recall $|D| = n_j$).

If there is an α approximation algorithm for UFL, then we are guaranteed to obtain a solution of cost at most $\alpha(f^{(j)}k + n_j)$. We set the value of $f^{(j)} = \gamma(n_j/k)$. Our total cost is at most $\alpha n_j(1 + \gamma)$.

We now consider a solution obtained by the approximation algorithm. Suppose we open $\beta_j k$ facilities at cost $f^{(j)}$ each. Suppose the number of clients that are within distance 1 of an open facility is $c_j n_j$. (Both β_j and c_j are ratios and not constants.) All other clients (uncovered elements) are at distance at least 3. The cost of this solution is at least $f^{(j)}\beta_j k + c_j n_j + 3(n_j - c_j n_j)$. This cost is at most $\alpha n_j(1 + \gamma)$. Thus we obtain (after dividing by n_j)

$$\alpha(1 + \gamma) \geq \gamma\beta_j + 3 - 2c_j \quad \Rightarrow \quad \alpha \geq \frac{\gamma\beta_j + 3 - 2c_j}{(1 + \gamma)}.$$

Define $c_{\beta_j} = (1 - 1/e^{(\beta_j/c)})$ for some fixed constant $0 < c < 1$. Now we end up with two cases:

Case a. For some j, $c_j \leq c_{\beta_j}$, then

$$\alpha \geq \frac{\gamma\beta_j + 3 - 2c_{\beta_j}}{(1 + \gamma)}.$$

If we fix γ and c, and compute the minimum value as a function of β_j we can compute the smallest value for the RHS. This will certainly be a lower bound for α, our approximation ratio. Taking the derivative with respect to β_j, tells us that the minimum is achieved at $\beta_j = c \ln(2/c\gamma)$. Substituting this value of β_j gives us

$$\alpha \geq \frac{1 + c\gamma}{1 + \gamma} + \frac{c\gamma}{1 + \gamma} \ln \frac{2}{c\gamma}.$$

Choosing $\gamma = 0.463$ gives $\alpha > 1.463$ when c is a constant close to 1.

Case b. For each j, $c_j > c_{\beta_j}$, then we have an algorithm that picks $\beta_j k$ sets and covers $c_j n_j$ elements each time we run the algorithm. Suppose in iteration j, the number of uncovered elements at the start of the iteration is n_j. Clearly, $n_{j+1} = n_j(1 - c_j)$ and $n_1 = |X|$. Suppose after ℓ iterations $n_{\ell+1} = 1$. The total number of sets that are picked is $\sum_{j=1}^{\ell} \beta_j k$, with an approximation factor of $\sum_{j=1}^{\ell} \beta_j$:

$$n_{\ell+1} = 1 = |X| \prod_{j=1}^{\ell} (1 - c_j),$$

$$\ln|X| = \sum_{j=1}^{\ell} \ln \frac{1}{1 - c_j}.$$

Using the fact that $c_j > (1 - 1/e^{(\beta_j/c)})$, we get for each j,

$$\beta_j < c \cdot \ln \frac{1}{1 - c_j},$$

$$\sum_{j=1}^{\ell} \beta_j < c \sum_{j=1}^{\ell} \ln \frac{1}{1 - c_j} = c \ln|X| \quad \text{for } c < 1.$$

Thus SET COVER $(X, \mathscr{S})$ is a $c \ln|X|$ approximation algorithm with $c < 1$. By Feige's result [4] this implies that $NP \subseteq DTIME[n^{O(\log \log n)}]$. ∎

Remark. Notice the above proof uses a distance metric where the distances are either 1 or 3. For such a distance metric of 1 or 3 and arbitrary costs we can develop an algorithm that achieves an approximation ratio of 1.463 (Corollary 4.4). For uniform costs and arbitrary distances we present an algorithm with an approximation factor of 2.225 (Subsection 4.4). It would be very interesting if we could use nonuniform costs and distances to get an improved lower bound for the general problem.

4. IMPROVED APPROXIMATION ALGORITHM

4.1. *Background Review*

We consider the linear programming relaxation of the integer program as discussed in Section 2. We relax the requirement that the x_{ij} and y_i values have to be integers. We now discuss our modified LP formulation and show how one can derive the algorithm by Shmoys, Tardos, and Aardal [16] using this LP formulation:

$$\min \sum_{i \in F} \sum_{j \in D} d_j c_{ij} x_{ij}$$

subject to

$$\sum_{i \in F} x_{ij} = 1 \qquad \text{for each } j \in D$$

$$x_{ij} \le y_i \qquad \text{for each } i \in F,\ j \in D$$

$$\sum_{i \in F} f_i y_i \le \lambda_f,$$

$$x_{ij} \ge 0 \quad \text{for each } i \in F,\ j \in D$$

$$y_i \ge 0 \quad \text{for each } i \in F.$$

This formulation assumes that we know the facility cost λ_f of the optimal integral solution. Since we do not know this value, we will "guess" it within a $(1 + \epsilon)$ factor by trying many different values. Assume that $f_i \ge 1$ for each $i \in F$. Let $\mathscr{F} = \sum_{i \in F} f_i$. Let $\epsilon > 0$. For each $\ell = 1 \cdots \lceil \log_{1+\epsilon} \mathscr{F} \rceil$, we will set $\lambda^\ell = (1 + \epsilon)^\ell$ and solve the above LP for by setting $\lambda_f = \lambda^\ell$ for each choice of ℓ. We will run our algorithm for each choice of ℓ and take the best solution. For some j, we have $\lambda^{j-1} < \lambda_f \le \lambda^j$. By using this value of j, we will accurately guess λ_f within a factor of $(1 + \epsilon)$ for any fixed ϵ.

We now outline the algorithm given by Shmoys, Tardos, and Aardal [16]. (Our description is a slight modification of the original description given in [16].) The algorithm works by fixing a parameter α between 0 and 1. We will refer to this algorithm as STA(α). Consider the optimal (fractional) solution corresponding to this new LP. The service cost of the optimal (fractional) solution is at most the service cost of the optimal of the integer program.

For each demand node j, order all the facilities by their distance from j in nondecreasing order. For an optimal (fractional) solution, we may assume that the demand node j exhausts a facility before routing demand from another facility at the same distance. This fact is based on the observation that an optimal solution will route demand from the closest open facilities. The formal argument is as follows: Redefine $x_{i_k j}$ as the following feasible solution to the LP formulation.

$$x_{i_k j} = \min\left(y_{i_k}, 1 - \sum_{s=1}^{k-1} y_{i_s} \right), \quad \text{when } \sum_{s=1}^{k-1} y_{i_s} < 1,$$

$$x_{i_k j} = 0, \qquad\qquad \text{otherwise}.$$

From the optimality of the original solution we can also conclude that the service cost of j remains the same under the new solution.

Define $N_\alpha(j)$ as the set of facilities (in the above order) such that their y_i values (and, hence, x_{ij} values due to the above property) sum to at least

α. That is $\sum_{s=1}^{i'} y_s \geq \alpha$ and $\sum_{s=1}^{i'-1} y_s < \alpha$. The threshold of j, denoted by $\tau(\alpha, j)$, is defined as the service distance to the farthest facility in the set $N_\alpha(j)$. Initially all demand nodes are unsatisfied and $\mathscr{D}$ is $\emptyset$. Recursively choose the unsatisfied demand node with the minimum $\tau(\alpha, j)$ value and add it to $\mathscr{D}$. Consider the cheapest facility $\min(j)$, in $N_\alpha(j)$ for this demand node. By opening a facility there, the cost goes up by at most a factor of $1/\alpha$. If for any node j', $N_\alpha(j) \cap N_\alpha(j')$ is nonempty, serve j' by the facility chosen by node j. Mark j and all such j' as satisfied.

It is easy to argue that every node j' gets served within service distance $3\tau(\alpha, j')$. For a node in $\mathscr{D}$ which was chosen with minimum $\tau(\alpha, j)$ value, it is trivially true. A node j' that was marked by j can be served within service distance $\tau(\alpha, j') + 2\tau(\alpha, j) \leq 3\tau(\alpha, j')$. This follows from the minimality of $\tau(\alpha, j)$.

Since each open facility corresponds to disjoint $N_\alpha(j)$, we will show that the cost of facilities is at most $(1/\alpha)\lambda_f$:

$$\sum_{j \in \mathscr{D}} f_{\min(j)} \leq \sum_{j \in \mathscr{D}} f_{\min(j)} \left(\frac{\sum_{i \in N_\alpha(j)} y_i}{\alpha} \right) \leq \frac{1}{\alpha} \sum_{j \in \mathscr{D}} \sum_{i \in N_\alpha(j)} f_i y_i \leq \frac{1}{\alpha} \lambda_f .$$

The service cost is at most $\sum_j 3 d_j \tau(\alpha, j)$.

Define

$$g(\alpha) = \frac{\sum_j d_j \tau(\alpha, j)}{\lambda_c} .$$

Thus the algorithm developed by the Shmoys, Tardos, and Aardal [16] guarantees a facility construction cost of at most $(1/\alpha)\lambda_f$ and a service cost of at most $3g(\alpha)\lambda_c$.

LEMMA 4.1. *Function g satisfies the property that* $\int_0^1 g(\alpha)\, d\alpha \leq 1$.

Proof. This is similar to the proof by Shmoys, Tardos, and Aardal [16]:

$$\int_0^1 g(\alpha)\, d\alpha = \sum_j \frac{d_j}{\lambda_c} \int_0^1 \tau(\alpha, j)\, d\alpha = \sum_j \left(\frac{d_j}{\lambda_c} \sum_i x_{ij} \cdot c_{ij} \right)$$

$$= \frac{1}{\lambda_c} \sum_j \text{cost for } j \text{ in fractional solution} \leq 1.$$

This establishes the claim. ∎

Observe that g is a monotonically increasing step function with at most $O(|F||D|)$ steps. (This will be used later to design a better algorithm.)

4.2. *The Greedy Improvement Algorithm*

By an (a, b) approximation we denote a solution with facility cost at most $a\lambda_f$ and service cost at most $b\lambda_c$. In [16] a $(1/\alpha, 3g(\alpha))$ approximation is used to give an algorithm with cost 3.16 times the optimal.

Assume we have an (a, b) approximation for the UFL problem. Assume that $b \geq 1 + \lambda_f/\lambda_c$. We provide a scheme that will give us a solution with a different approximation guarantee. *The key idea is to open more facilities to reduce the service cost*.

Let $S(F)$ denote the minimum service cost corresponding to a set F of facilities,

$$S(F) = \sum_{j \in D} d_j \cdot \text{shortest path from } j \text{ to a member of } F.$$

GREEDY IMPROVEMENT ALGORITHM (α).

1. F = Initial Set of Facilities formed by STA(α)
2. **repeat**
3. Consider an unused facility v
4. Let $Gain(v) = S(F) - S(F \cup \{v\}) - f_v$
5. Find the facility v' with the highest (positive) $Gain(v')/f_{v'}$ value
6. $F = F \cup \{v'\}$
7. **until** no facility has positive gain
8. Return F as the set of facilities
9. **end-proc**

We now present the full algorithm. The approximation factor is shown in Theorem 4.5.

FINAL ALGORITHM(λ_f).

Step 1. Solve the LP formulated in Subsection 4.1.
Notice that $g(\alpha)$ is a step function with polynomially many steps.

Step 2. For each α where the function $g(\alpha)$ has a step, run GREEDY IMPROVEMENT ALGORITHM(α).

Step 3. Choose the solution which minimizes the cost in Step 2.

4.3. *Proof of Approximation Factor*

THEOREM 4.2. *The total cost* (*facility and service*) *corresponding to the set F obtained by* GREEDY IMPROVEMENT ALGORITHM (α) *is at most* $\lambda_f(a + \ln((b-1)/(\lambda_f/\lambda_c)) + 1) + \lambda_c$.

Proof. At the start of iteration i let the set of facilities be F_i. Suppose for an iteration i the service cost $S(F_i)$ exceeds $\lambda_c + \lambda_f$. If we add the facilities used by the optimal solution, the service cost reduces to λ_c. Thus there exists a set of facilities whose total cost is λ_f and total gain of these facilities is $S(F_i) - \lambda_c - \lambda_f$.

LEMMA 4.3. *At the start of iteration i, there exists a facility whose gain to cost ratio is at least $(S(F_i) - \lambda_c - \lambda_f)/\lambda_f$.*

Proof. Suppose each facility has a gain to cost ratio that is strictly smaller than $\Gamma = (S(F_i) - \lambda_c - \lambda_f)/\lambda_f$. Let OPT_F be an optimal set of facilities, with total facility cost λ_f and service cost λ_c.

If we add a facility j in OPT_F to the current set of facilities F_i, let the gain be g_j. Notice that if we add facility j to a set of facilities that contain F_i then the gain is only smaller. Clearly, $g_j < \Gamma \cdot f_j$, where f_j is the cost of facility j. Suppose we add the facilities in OPT_F one at a time (in increasing facility number). Let $\Delta S(F_i^j)$ be the change of the service cost when facility j is added to $F_i \cup \{1, \dots, j-1\}$. By definition of g_j, $\Delta S(F_i^j) - f_j \le g_j < \Gamma \cdot f_j$. Thus,

$$\Delta S\left(F_i^j\right) < (1 + \Gamma) f_j.$$

By summing (over all j) and replacing the value of Γ we obtain

$$\sum_{j \in OPT_F} \Delta S\left(F_i^j\right) < \left(S(F_i) - \lambda_c\right) \frac{\sum_{j \in OPT_F} f_j}{\lambda_f} \le S(F_i) - \lambda_c.$$

This is a contradiction, since the service cost drops by at least $S(F_i) - \lambda_c$ when we add all the facilities in OPT_F. ∎

This tells us that we stop only after our service cost has decreased below $\lambda_c + \lambda_f$. Suppose we *cross this threshold at iteration* ℓ. Using the fact that the total cost only decreases, it is sufficient to show that the theorem holds at the end of the iteration ℓ.

Let us denote the cost of the set of facilities F_i by $C(F_i)$. Since we chose the facility v' with the maximum gain to cost ratio,

$$\begin{aligned} \frac{S(F_i) - S(F_{i+1})}{C(F_{i+1}) - C(F_i)} &= \frac{\mathrm{Gain}(v') + f(v')}{f(v')} \\ &\ge \frac{S(F_i) - \lambda_c - \lambda_f}{\lambda_f} + 1 = \frac{S(F_i) - \lambda_c}{\lambda_f}. \end{aligned}$$

Rearrange the last equation as

$$S(F_{i+1}) \leq S(F_i) - \big(S(F_i) - \lambda_c\big)\left(\frac{C(F_{i+1}) - C(F_i)}{\lambda_f}\right). \tag{1}$$

Subtracting λ_c from both sides, and observing that $1 - x \leq e^{-x}$ for $x \leq 1$,

$$\begin{aligned} S(F_{i+1}) - \lambda_c &\leq \big(S(F_i) - \lambda_c\big)\left(1 - \frac{C(F_{i+1}) - C(F_i)}{\lambda_f}\right) \\ &\leq \big(S(F_i) - \lambda_c\big) \cdot e^{-(C(F_{i+1}) - C(F_i))/\lambda_f}. \end{aligned}$$

The solution to this recurrence, for any j, is

$$S(F_j) \leq \lambda_c + \big(S(F_1) - \lambda_c\big)e^{-(C(F_j) - C(F_1))/\lambda_f}.$$

Notice that $C(F_1) \leq a\lambda_f$ and $S(F_1) \leq b\lambda_c$. After $C(F)$ exceeds $C(F_1) + \lambda_f \cdot \ln((b-1)\lambda_c/\lambda_f)$, the service cost is guaranteed to decrease below $\lambda_c + \lambda_f$.

However, this does not give the approximation bound desired, because $C(F)$ *could have exceeded* $C(F_1) + \lambda_f \cdot \ln((b-1)\lambda_c/\lambda_f)$. This situation occurs at iteration ℓ, and if we exceed this, then the distance cost also does go down accordingly.

Since $S(F_\ell) > \lambda_c + \lambda_f$, rewrite this for some $x > 1$,

$$S(F_\ell) = \lambda_c + x\lambda_f.$$

Let the facility cost paid in the last iteration be $\lambda_f \Delta_c$. (Note that $\Delta_c \leq 1$, since we can ignore facilities whose cost exceeds λ_f.) From Eq. (1) we conclude that $S(F_{\ell+1}) \leq S(F_\ell) - \Delta_c(S(F_\ell) - \lambda_c)$. Rewrite this as for some $\mu \geq 1$,

$$S(F_{\ell+1}) = S(F_\ell) - \mu\Delta_c\big(S(F_\ell) - \lambda_c\big) = \lambda_c + \lambda_f x(1 - \mu\Delta_c).$$

Since $S(F_{\ell+1}) \leq \lambda_c + \lambda_f$, we get the relation between x, Δ_c, and μ:

$$x(1 - \mu\Delta_c) \leq 1 \quad \Leftrightarrow \quad -\frac{1}{x} + 1 - \mu\Delta_c \leq 0.$$

The facility cost paid at the earlier iteration can be bounded as

$$\begin{aligned} C(F_\ell) &\leq C(F_1) + \lambda_f \cdot \ln\frac{S(F_1) - \lambda_c}{S(F_\ell) - \lambda_c} \\ &\leq \left(a + \ln\frac{b-1}{x\dfrac{\lambda_f}{\lambda_c}}\right)\lambda_f. \end{aligned}$$

Thus the total cost paid by us is at most $C(F_\ell) + S(F_{\ell+1}) + \lambda_f \Delta_c$, which is at most

$$\left(a + \ln \frac{b-1}{\dfrac{\lambda_f}{\lambda_c}} \right) \lambda_f + \lambda_c + \lambda_f(-\ln x + x - \mu x \Delta_c + \Delta_c).$$

The derivative w.r.t. x of the third term in the summand is, $-1/x + 1 - \mu\Delta_c$. Notice that this is negative, due to relations between x, Δ_c, and μ. The term therefore is greatest when x approaches 1, but is $1 - (\mu - 1)\Delta_c$, and, hence, at most 1. This proves the lemma. ∎

COROLLARY 4.4. *For a metric with distances* 1 *and* 3 *we have a* 1.463 *approximation.*

Proof. We may assume that the optimal solution opens at least k facilities for any constant k (otherwise by enumeration over all subsets of size at most k, we can obtain the optimal solution). If we open a single facility, we get $a \leq 1/k$ and, since all demands are covered at distance 3, we get $b \leq 3$. By applying a greedy improvement scheme and by using Theorem 4.2, we obtain a solution with cost at most $\lambda_f(a + \ln((b-1)/(\lambda_f/\lambda_c)) + 1) + \lambda_c \leq \alpha(\lambda_f + \lambda_c)$. Simplifying (and setting $\gamma = \lambda_f/\lambda_c$) gives

$$\alpha \geq 1 + \frac{1}{\left(1 + \dfrac{1}{\gamma}\right)k} + \frac{\gamma}{\gamma + 1} \ln \frac{2}{\gamma}.$$

Since we may assume that k is at least some fixed constant, by taking the maximum over all $\gamma > 0$ we get $\alpha \geq (1.463 + \epsilon)$. Thus we can choose $\alpha = 1.463 + \epsilon$ for any fixed $\epsilon > 0$. ∎

THEOREM 4.5. *The above algorithm yields a solution of cost at most* 2.408 *times the optimal.*

Proof. Let $c = 2.408$ and assume the contrary. Let $\gamma = \lambda_f/\lambda_c$. For an $(1/\alpha, 3g(\alpha))$ approximation, the facility cost is at most $\lambda_f/\alpha = \lambda_c\gamma/\alpha$ and the service cost is at most $3g(\alpha)\lambda_c$.

If for any value of α tried by the algorithm, we have $3g(\alpha)\lambda_c \leq \lambda_c(c(\gamma+1) - \gamma/\alpha)$, we immediately obtain a factor c approximation. (Since $(1/\alpha)\lambda_f + 3g(\alpha)\lambda_c \leq (\gamma/\alpha)\lambda_c + \lambda_c(c(\gamma+1) - \gamma/\alpha) \leq c\lambda_c(\gamma+1) \leq cOPT$.) Thus w.l.o.g.,

$$3g(\alpha) > c(1+\gamma) - \frac{\gamma}{\alpha}.$$

Since $g(\alpha)$ is a monotonic step function, the inequality is valid for every α where a step occurs, we can extend the inequality to hold for all α. This inequality is only meaningful for $\alpha \geq \gamma/(c(1+\gamma))$.

When $\alpha \geq \gamma/((c-1)(1+\gamma))$, we have $3g(\alpha) > (1+\gamma)$. For this value of b, we can use the greedy algorithm to get a cost of at most

$$\lambda_c\left(\gamma a + \gamma \ln \frac{3g(\alpha)-1}{\gamma} + 1 + \gamma\right).$$

The above quantity is more than $c(1+\gamma)\lambda_c$, or else we would have a factor c approximation. Using the step function property, this gives us for $\alpha \geq \gamma/((c-1)(1+\gamma))$,

$$3g(\alpha) > 1 + \gamma e^{(c-1)(1+1/\gamma)-1/\alpha}.$$

Now integrating over the interval of $0 \leq \alpha \leq 1$,

$$\begin{aligned}
\int_0^1 3g(\alpha)\,d\alpha &\geq \int_{\gamma/c(1+\gamma)}^{\gamma/(c-1)(1+\gamma)} 3g(\alpha)\,d\alpha + \int_{\gamma/(c-1)(1+\gamma)}^{1} 3g(\alpha)\,d\alpha \\
&> \int_{\gamma/c(1+\gamma)}^{\gamma/(c-1)(1+\gamma)} \left(c(1+\gamma) - \frac{\gamma}{\alpha}\right) d\alpha \\
&\quad + \int_{\gamma/(c-1)(1+\gamma)}^{1} \left(1 + \gamma e^{(c-1)(1+1/\gamma)-1/\alpha}\right) d\alpha \\
&\geq \frac{\gamma}{c-1} - \gamma \ln \frac{c}{c-1} + 1 - \frac{\gamma}{(c-1)(1+\gamma)} \\
&\quad + \gamma e^{(c-1)(1+\gamma)/\gamma} \int_{\gamma/(c-1)(1+\gamma)}^{1} e^{-1/\alpha}\,d\alpha
\end{aligned}$$

Call the function on the RHS as $\Gamma(c,\gamma)$. We can verify that $\Gamma(c,\gamma) > 3.0012$ for all positive γ. The guessing of the facility cost would introduce a factor of $(1+\epsilon)$ in the term involving $1/\alpha$. Notice that for suitably small ϵ, this gives us a contradiction because $\int_0^1 3g(\alpha)\,d\alpha$ is at most 3 by Lemma 4.1. ▮

4.4. *Uniform Facility Cost*

If the facility costs are uniform, we can change the way we obtain an integral solution and obtain an approximation factor of 2.225. The central theme of this section relies on the fact that if all the facilities are of the same cost, it is most profitable to open a facility nearest to the demand points. We define the neighborhoods $N_\alpha(j)$ and the function $\tau(\alpha, j)$ as before. Let $\tau(0+, j)$ denote the distance to the closest facility supplying j, or $\tau(0+, j) = \min_i\{c_{ij} | x_{ij} > 0\}$. The quantity $g(0+)$ is defined analogous to

$g(\alpha)$ as $(\sum_{j \in D} d_j \tau(0+, j))/\lambda_c$. In each iteration pick the node with the smallest $\tau(\alpha, j) + \tau(0+, j)$.

If the demand node achieving the minimum is j', open a facility at the location closest to j', where a facility can be opened. Observe that the distance from that facility to j' is at most $\tau(0+, j')$. So all nodes v whose $N_\alpha(v)$ intersects the set $N_\alpha(j')$ would now incur a cost of $\tau(\alpha, v) + \tau(\alpha, j') + \tau(0+, j')$. By the choice of j', this is at most $2\tau(\alpha, v) + \tau(0+, v)$. Summing over all v, we get that the service cost is at most $2g(\alpha) + g(0+)$. Once again, the integral of $g(\alpha)$ is the optimal fractional service cost and at most 1. Notice that $g(0+) \leq 1$, else the integral of g will certainly exceed 1.

By the same reasoning as in the proof earlier, if the distance cost exceeds $1 + \gamma$, we can use a greedy improvement to lower it. We again assume that we do not achieve an approximation factor of c:

$$\begin{aligned} 2g(\alpha) + g(0+) &\geq 1 + \gamma e^{(c-1)(1+1/\gamma) - 1/\alpha} \quad \text{for } \alpha \geq u = \frac{\gamma}{(c-1)(1+\gamma)} \\ &\geq 3g(0+) \quad \text{for all } \alpha. \end{aligned}$$

Integrating both sides using the first inequality over $(u, 1)$ and the second over $(0, u)$,

$$2 + g(0+) \geq 1 - u + \int_u^1 \gamma e^{1/u - 1/\alpha} \, d\alpha + 3ug(0+).$$

Set $c = 2.225$. We have two cases depending on the value of u. If $u \geq \frac{1}{3}$ we have $\gamma \geq (c-1)/(4-c)$ and

$$2 \geq 1 - u + \int_u^1 \gamma e^{1/u - 1/\alpha} \, d\alpha + (3u - 1)g(0+).$$

Since $(3u - 1) \geq 0$ we get

$$2 \geq 1 - u + \int_u^1 \gamma e^{1/u - 1/\alpha} \, d\alpha.$$

We can verify that for all values of $u \geq \frac{1}{3}$ the RHS exceeds the LHS.

If $u < \frac{1}{3}$ we get

$$2 \geq (1 - u)(1 - g(0+)) + 2ug(0+) + \int_u^1 \gamma e^{1/u - 1/\alpha} \, d\alpha.$$

Since $(1 - u) > 2u$ we get

$$2 \geq 2u(1 - g(0+)) + 2ug(0+) + \int_u^1 \gamma e^{1/u - 1/\alpha} \, d\alpha.$$

Simplifying, we get

$$2 \geq 2u + \int_u^1 \gamma e^{1/u - 1/\alpha}\, d\alpha.$$

We can verify that for this value of c, and the related range of γ, the RHS is always greater than LHS.

5. GAP BETWEEN FRACTIONAL AND INTEGRAL SOLUTION

One immediate question that comes to mind on reading the paper by Shmoys, Tardos, and Aardal [16] is the issue of the "gap" between the integral and fractional solution. They show that they can round a fractional solution to an integral solution by increasing the total objective function cost by a factor of 3.16. One might ask—what is the inherent limitation of this method? What is the best constant one can hope for by this technique?

We give a family of examples for which any rounding strategy applied to the LP formulation in [1, 16] will be forced to increase the cost of the integral solution by a factor of about 1.463.

The set F is a set of k vertices. Each $f_i = 1$. The set D is a set of demand points, with each $d_j = 1$. We create $\binom{k}{\ell}$ demand points, one corresponding to *each* subset $S \subset F$ of size ℓ. This demand point is at a service distance c from the facilities in subset S. One should note that the distance to a facility $j \in F - S$ is exactly $3c$. (There must be a demand point that is adjacent to j and a facility in S.)

The cost of the minimum fractional solution to the linear programming formulation [1, 16] is at most $k/\ell + c \cdot \binom{k}{\ell}$. (This solution is obtained by assigning a fractional value y_i at each facility of $1/\ell$.) Since each demand point is adjacent to a set S of facilities, it can be fractionally satisfied with cost c.

The optimal integral solution has cost

$$\min_{k'} \left(k' + c\binom{k}{\ell} + 2c\binom{k-k'}{\ell} \right).$$

This cost is obtained by choosing k' facilities from the set F, and $\binom{k-k'}{\ell}$ is the number of demand points at distance $3c$ from the set of open facilities. Our ratio r is at least

$$\frac{\min_{k'}\left(k' + c\left(\binom{k}{\ell} + 2\binom{k-k'}{\ell} \right) \right)}{\frac{k}{\ell} + c \cdot \binom{k}{\ell}}.$$

Let $xc\binom{k}{\ell} = k/\ell$, where $x = 0.463$. Let $\delta = 1 + 1/\sqrt{\ell}$.

There are two possible cases:

Case (a). $k' \geq 2((\delta - 1)/\delta^2)(k - \ell)$. In this case we can say that

$$r \geq \frac{k'}{\frac{k}{\ell}\left(1 + \frac{1}{x}\right)} \geq \frac{2\ell x(\delta - 1)}{\delta^2}\left(\frac{k - \ell}{k}\right)\frac{1}{(1 + x)}.$$

Since we can choose ℓ such that $1 - \ell/k > \frac{1}{2}$ and $\delta^2 \leq 2$,

$$r \geq \frac{x\sqrt{\ell}}{2(1 + x)} \geq 1.5.$$

Case (b). $k' < 2((\delta - 1)/\delta^2)(k - \ell)$. This can be rewritten as $\frac{1}{2}(k'\delta/(k - \ell))^2 < (\delta - 1)(k'/(k - \ell))$. Using the fact that $e^{-z} \leq 1 - z + \frac{1}{2}z^2$, we obtain

$$e^{-\delta k'/(k-\ell)} \leq 1 - \frac{\delta k'}{k - \ell} + \frac{1}{2}\left(\frac{\delta k'}{k - \ell}\right)^2.$$

Using the bound $\frac{1}{2}(k'\delta/(k - \ell))^2 < (\delta - 1)(k'/(k - \ell))$, we obtain

$$e^{-\delta k'/(k-\ell)} \leq 1 - \frac{\delta k'}{k - \ell} + \frac{1}{2}\left(\frac{\delta k'}{k - \ell}\right)^2 < 1 - \frac{k'}{k - \ell}.$$

Let

$$\nu = \frac{\binom{k - k'}{\ell}}{\binom{k}{\ell}} = \prod_{i=1}^{i=\ell-1}\left(1 - \frac{k'}{k - i}\right) > \prod_{i=0}^{i=\ell-1}\left(1 - \frac{k'}{k - \ell}\right)$$

$$> \left(e^{-\delta k'/(k-\ell)}\right)^{\ell}.$$

Thus

$$r \geq \frac{\min_{k'}\left(k' + c\left(\binom{k}{\ell} + 2\nu\binom{k}{\ell}\right)\right)}{k/\ell + c \cdot \binom{k}{\ell}}$$

$$> \frac{\min_{k'}\left(k' + c\binom{k}{\ell}(1 + 2e^{-\delta k'\ell/(k-\ell)})\right)}{k/\ell + c \cdot \binom{k}{\ell}}.$$

Let $\beta = (\ell/k)k'$. We obtain

$$r \geq \min_{\beta} \frac{\beta x c\binom{k}{\ell} + c\binom{k}{\ell}(1 + 2e^{-\beta\delta k/(k-\ell)})}{c\binom{k}{\ell}(1+x)}$$

$$= \min_{\beta} \frac{\beta x + 1 + 2e^{-\beta\delta(k/(k-\ell))}}{1+x}.$$

This is minimized for $\beta = ((k-\ell)/k\delta)\ln(2\delta k/(x(k-\ell)))$. This yields

$$r \geq \frac{1 + \dfrac{(k-\ell)x}{k\delta} + \dfrac{(k-\ell)x}{k\delta}\ln\dfrac{2\delta k}{x(k-\ell)}}{1+x}.$$

In the limit when $(k-\ell)/k\delta \to 1$, the supremum is $1 + x\ln(2/x)/(1+x)$ which is maximized for the given choice of $x = 0.463$ to be 1.463. The maximum value is obtained for $x = 0.463$ when we get $r \geq 1.463$.

6. COMPLETE METRIC FACILITY LOCATION

Recall that in this version of the problem $F = D = N$.

6.1. *Lower Bound*

We use the same technique as before to show a hardness of 1.278. However, instead of the set cover, we will use the minimum dominating set problem. The *dominating set* problem asks for a smallest subset of vertices such that each vertex either belongs to the subset, or is adjacent to a vertex in the subset. Set cover reduces to this problem by an approximation preserving reduction, and it is also hard to approximate this problem within a factor $c'\ln n$ for any $c' < 1$, unless $NP \not\subseteq DTIME[n^{O(\log\log n)}]$.

Using a similar line of argument as in Subsection 3.2, we can argue that if the size of a minimum dominating set is k, choosing βk nodes (facilities) cannot dominate more than $1 - e^{-\beta}$ fraction of the nodes. The rest of the nodes would be at least at distance 2 from any facility. Choosing $f \cdot k = \gamma n$, the cost of solution is at least

$$\gamma\beta n + \left(1 - \frac{1}{e^{\beta}}\right)n + \frac{2}{e^{\beta}}n.$$

Since the minimum dominating set guarantees a solution of cost at most $(1 + \gamma)n$. Minimizing over β the ratio is at least

$$1 + \frac{\gamma \ln(1/\gamma)}{1 + \gamma}$$

which exceeds 1.278 for $\gamma = 0.28$.

6.2. *Upper Bound for Uniform Costs*

In this section we show a 2.104 approximation for the special case when all facility costs are identical, and $F = D = N$. We need to modify the rounding scheme proposed in [16]. We define an LP in the same way as before. We can guess exactly how many facilities the optimal solution picks. For every node j, order the facilities $\{i\}$ that serve node j in increasing order of service distance c_{ij}. For every α, define $N_\alpha(j)$ as the set of facilities closest to j and serving j such that their y_i values sum to at least α. The threshold of j, denoted by $\tau(\alpha, j)$, is defined as the service distance to the farthest facility in the set $N_\alpha(j)$,

$$\begin{aligned} d_j \int_0^1 \tau(\alpha, j)\, d\alpha &= d_j \sum_i x_{ij} \cdot c_{ij} \\ &= \text{service cost of } j \text{ in fractional solution.} \end{aligned}$$

Recursively choose the unsatisfied node with the minimum $\tau(\alpha, j)$ value and place a facility there. If for any node j', $N_\alpha(j) \cap N_\alpha(j')$ is nonempty, serve j' by node j. Mark j and all such j' as satisfied.

It is easy to argue that each node j' gets served within service distance $2\tau(\alpha, j')$. Also notice that since each facility placed corrresponds to disjoint $N_\alpha(j)$, the cost of facilities increase by at most $1/\alpha$. Thus this gives an $(1/\alpha, 2g'(\alpha))$ approximation, where $g'(\alpha) = \sum_j d_j \tau(\alpha, j)/\lambda_c$. The relation which holds on g' is

$$\int_0^1 g'(\alpha)\, d\alpha \le 1.$$

Now using this and the technique of the previous algorithm if there is no c approximation, we get twice the integral above is $\Gamma(c, \gamma)$. If $c = 2.104$, $\Gamma(c, \gamma)$ exceeds 2 for all γ. Hence, the result follows.

ACKNOWLEDGMENTS

We thank Neal Young and Uri Feige for discussions related to the hardness bound. We thank Randeep Bhatia, Moses Charikar, Rajeev Motwani, and Yoram Sussmann for useful discussions. We thank David Shmoys for providing us a copy of [16]. We thank Chandra Chekuri and the referees for useful comments on an earlier draft of this paper.

REFERENCES

1. M. L. Balinski, On finding integer solution to linear programs, *in* "Proc. of the IBM Scientific Computing Symposium on Combinatorial Problems, 1966," pp. 225–248.
2. G. Cornuejols, G. L. Nemhauser and L. A. Wolsey, The uncapacitated facility location problem, *in* "Discrete Location Theory, 1990, (Mirchandani and Francis, Eds.), pp. 119–171.
3. F. Chudak, Improved approximation algorithms for uncapacitated facility location, *in* "6th Conference on Integer Programming and Combinatorial Optimization," LNCS, Vol. 1412, 180–194, Springer Verlag, New York/Berlin, 1998.
4. U. Feige, A threshold of ln n for approximating set-cover, *in* "28th ACM Symposium on Theory of Computing, 1996," pp. 314–318.
5. J. Håstad, Some optimal inapproximability results, *in* "29th ACM Symposium on Theory of Computing, 1997," 1–10.
6. D. S. Hochbaum, Heuristics for the fixed cost median problem, *Math. Programming* **22** (1982), 148–162.
7. D. S. Hochbaum (Ed.), "Approximation Algorithms for NP-Hard Problems," PWS, Boston, MA, 1996.
8. M. Korupolu, C. G. Plaxton, and R. Rajaraman, Analysis of a local search heuristic for facility location problems, *in* "9th ACM-SIAM Annual Symposium on Discrete Algorithms, 1998," 1–10.
9. A. A. Kuehn and M. J. Hamburger, A heuristic program for locating warehouses, *Manag. Sci.* **9** (1963), 643–666.
10. J.-H. Lin and J. S. Vitter, "Approximation algorithms for geometric median problems, *Inform. Process. Lett.* **44** (1992), 245–249.
11. C. Lund and M. Yannakakis, On the hardness of approximating minimization problems, *Assoc. Comput. Mach.* **41**(5) (1994), 960–981.
12. A. S. Manne, Plant location under economies-of-scale-decentralization and computation, *Manage. Sci.* **11** (1964), 213–235.
13. A. Mirzain, Langrangian relaxation for the star-star concentrator location problem, *Networks* **15** (1985), 1–20.
14. J. M. Mulvey and H. L. Crowder, Cluster analysis: An application of Lagrangian relaxation, *Manage. Sci.* **25** (1979), 329–340.
15. C. H. Papadimitriou and M. Yannakakis, Optimization, approximation, and complexity classes, *J. Comput. System Sci.* **43** (1991), 425–440.
16. D. Shmoys, E. Tardos, and K. Aardal, Approximation algorithms for facility location problems, *in* "29th ACM Symposium on Theory of Computing, 1997," 265–274.
17. J. F. Stollsteimer, A working model for plant numbers and locations, *J. Farm Econom.* **45** (1963), 631–645.