

---

Faculty of Mathematical Sciences

University of Twente

University for Technical and Social Sciences

---

---

P.O. Box 217

7500 AE Enschede

The Netherlands

Phone: +31-53-4893400

Fax: +31-53-4893114

Email: [memo@math.utwente.nl](mailto:memo@math.utwente.nl)

---

MEMORANDUM No. 1574

A simple dual ascent algorithm  
for the multilevel facility location problem

A.F. BUMB AND W. KERN

MARCH 2001

ISSN 0169-2690

# A simple dual ascent algorithm for the multilevel facility location problem

A.F. Bumb and W.Kern  
Faculty of Mathematical Sciences  
University of Twente

March 9, 2001

## Abstract

We present a simple dual ascent method for the multilevel facility location problem which finds a solution within 6 times the optimum for the uncapacitated case and within 12 times the optimum for the capacitated one. The algorithm is deterministic and based on the primal-dual technique.

**Keywords:** approximation algorithms, linear programming, discrete location

**AMS Subject Classifications:** 68W25, 90C05, 90B80

## 1 Introduction

An important problem in facility location is to select a set of facilities, such as warehouses or plants, in order to minimize the total cost of opening facilities and of satisfying the demands for some commodity (see Cornuejols, Nemhauser & Wolsey [CNW90]). There are two models which are mainly studied in the literature: *the uncapacitated facility location model* (UFL), in which each facility can serve an unlimited number of clients and *the capacitated facility location model* in which each facility can serve at most a certain number of clients. We shall briefly survey the results known on approximation algorithms for these two facility location problems.

Throughout this paper, a  $\rho$ -*approximation algorithm* is a polynomial time algorithm that always finds a feasible solution with objective function value within  $\rho$  times the optimum. The value  $\rho$  is called the *performance guarantee* of the algorithm.

It is known that the metric uncapacitated facility location problem (the service costs are symmetric and satisfy the triangle inequality) is NP-hard (see [CNW90]). Independently, Guha & Khuller [GK98] and Sviridenko [S97] have shown that it is Max SNP-hard. They have also proved that the existence of a  $\rho$ -approximation algorithm with  $\rho < 1.46$  would imply that  $P = NP$ .

The first constant-factor approximation algorithm for the metric uncapacitated facility location problem was developed by Shmoys, Tardos & Aardal [STA97], and was based on the filtering technique introduced by Lin & Vitter [LV92]. The performance guarantee of this algorithm is 3.16. Their result was subsequently improved by Guha & Khuller [GK98], who obtained a 2.41-approximation algorithm applying a greedy procedure to the solution obtained by [STA97] and by Chudak & Shmoys [C98], who provided an 1.73-approximation algorithm based on a randomized rounding technique, in which the variables are rounded in a dependent way. For the capacitated version, Chudak & Shmoys [CS99] obtained a 3-approximation algorithm in the case that one can open at one location an infinite number of copies of one facility. The drawback of these algorithms is that they need to solve large linear programs and therefore they are not practical for many applications.

Recently, Jain & Vazirani proposed the first combinatorial approximation algorithm, based on the primal-dual technique, which achieves the performance guarantee of 3 for the uncapacitated version and

4 for the capacitated one. Charikar & Guha [CG99] observed that scaling the costs, running the primal-dual algorithm for the new problem and improving the solution using the greedy procedure from [GK98] can reduce the performance guarantee to 1.85.

The multilevel extension of the UFL problem has also been extensively studied in the literature (see [A96]). In this problem there are  $k$  types of facilities to be built: one type of depots and  $(k - 1)$  types of transit stations. Each unit of demand must be shipped from a depot through transit stations of type  $k - 1, \dots, 1$  to the demand points. Being an extension of UFL, this problem is Max SNP-hard as well.

In [STA97] Shmoys, Tardos & Aardal extended their algorithm for one level to multiple levels of facilities, obtaining a performance guarantee of 3.16, constant which was later improved to 3 by Aardal, Chudak & Shmoys [ACS99]. As in the one level case, their algorithm is based on a randomized rounding of a LP solution to an integer one. The first combinatorial algorithm for the multilevel facility location problem was developed by Meyerson, Munagala & Plotkin [MMP], and finds a solution within  $O(\log |D|)$  the optimum, where  $D$  is the set of demand points.

Using an idea from [JV99], we present a simple greedy (dual ascent) method for the multilevel facility location problem that finds a solution within 6 times the optimum. The algorithm can be easily extended to the capacitated case, when each facility can serve only a certain number of demand points. The approximation guarantee for this case is 12.

## 2 The metric multilevel uncapacitated facility location problem

Consider a complete  $(k + 1)$ -partite graph  $G = (V, E)$  with  $V = V_0 \cup \dots \cup V_k$  and  $E = \bigcup_{l=1}^k V_{l-1} \times V_l$ . The set  $D = V_0$  is the set of *demand nodes* and  $F = V_1 \cup \dots \cup V_k$  is the set of possible *facility locations* (at level 1, ...,  $k$ ). We are given *edge costs*  $c \in R_+^E$  and *opening costs*  $f \in R_+^F$  (i.e., opening a facility at  $i \in F$  incurs a cost  $f_i \geq 0$ ). We assume that  $c$  is induced by a metric on  $V$ . Without loss of generality we can assume that there are no edges of cost 0.

**REMARK** Our results also hold in a slightly more general setting, where we require only for  $e \in V_0 \times V_1$  that  $c(e) \leq c(p)$  for any path  $p$  joining the endpoints of  $e$ .

Denote by  $P$  the set of paths of length  $k - 1$  joining some node in  $V_1$  to some node in  $V_k$ . If  $j \in D$  and  $p = (v_1, \dots, v_k) \in P$ , we let  $jp$  denote the path  $(j, v_1, \dots, v_k)$ . As usual  $c(p)$  and  $c(jp)$  denote the length of  $p$  resp.  $jp$  (with respect to  $c$ ).

The corresponding *UFL* problem can now be stated as follows: Determine for each  $j \in D$  a path  $p_j \in P$  (along "open facilities") so as to minimize

$$\sum_{j \in D} c(jp_j) + f\left(\bigcup_{j \in D} p_j\right).$$

**REMARK** In this setting we assume that each  $j \in D$  has a demand of one unit to be shipped along  $p_j$ . Our results easily extend to arbitrary positive demands.

To derive an integer programming formulation of the UFL problem, we introduce the 0 - 1 variables  $y_i$  ( $i \in F$ ) to indicate whether  $i \in F$  is open and the 0 - 1 variables  $x_{jp}$  ( $j \in D, p \in P$ ) to indicate whether  $j$  is served along  $p$ .

We let

$$c(x) := \sum_{p \in P} \sum_{j \in D} c_{jp} x_{jp}$$

and

$$f(y) := \sum_{i \in F} f_i y_i.$$

The UFL problem is now equivalent to

$$\begin{aligned}
& \text{minimize } c(x) + f(y) \\
& \text{subject to } \sum_{p \in P} x_{jp} = 1, & \text{for each } j \in D & \quad (1) \\
(P_{int}) \quad & \sum_{p \ni i} x_{jp} \leq y_i, & \text{for each } i \in F, j \in D & \quad (2) \\
& x_{pj} \in \{0, 1\}, & \text{for each } p \in P, j \in D \\
& y_i \in \{0, 1\}, & \text{for each } i \in F
\end{aligned}$$

Constraints (1) ensure that each  $j$  gets connected via some path and constraints (2) ensure that the paths only use open facilities.

The  $LP$ -relaxation of  $(P_{int})$  is given by

$$\begin{aligned}
(P) \quad & \text{minimize } c(x) + f(y) \\
& \text{subject to (1), (2)} \\
& x_{jp} \geq 0 \\
& y_i \geq 0.
\end{aligned}$$

Note that  $x_{jp} \leq 1$  is implied by (1) and  $y_i \leq 1$  holds automatically for any optimal solution  $(x, y)$  of  $(P)$ .

The standard way of proving a 0 – 1 solution  $(x, y)$  of  $(P_{int})$  to be a  $\rho$ -approximation is to show that

$$c(x) + f(y) \leq \rho C_{LP} \quad (2.1)$$

where  $C_{LP}$  is the optimum value of  $(P)$ .

### 3 The primal-dual algorithm

The basic idea of the primal-dual approach is to exhibit a primal 0 – 1 solution  $(x, y)$  satisfying (2.1) by considering the dual of  $(P)$ . Introducing dual variables  $v_j$  and  $t_{ij}$  corresponding to constraints (1) and (2) in  $(P)$ , the dual becomes

$$\begin{aligned}
& \text{maximize } \sum_{j \in D} v_j \\
& v_j - \sum_{i \in p} t_{ij} \leq c(jp), \text{ for each } p \in P, j \in D & \quad (3)
\end{aligned}$$

$$\begin{aligned}
& \sum_{j \in D} t_{ij} \leq f_i, \text{ for each } i \in F & \quad (4) \\
& t_{ij} \geq 0, \text{ for each } i \in F, j \in D
\end{aligned}$$

Intuitively, the dual variable  $v_j$  indicates how much  $j \in D$  is willing to pay for getting connected. The value of  $t_{ij}$  indicates how much  $j \in D$  is willing to contribute to the opening cost  $f_i$  (if he would be connected along a path through  $i$ ).

We aim at constructing a primal feasible 0 – 1 solution  $(x, y)$  and a feasible dual solution  $(v, t)$  such that

$$c(x) + f(y) \leq 6 \sum_{j \in D} v_j,$$

implying (2.1) for  $\rho = 6$ .

We first describe how to construct the dual solution  $(v, t)$ . To this end, we introduce the following notation w.r.t. an arbitrary feasible solution  $(v, t)$  of  $(D)$ :

A facility  $i \in F$  is *fully paid* when

$$\sum_{j \in D} t_{ij} = f_i. \quad (3.1)$$

A demand point  $j \in D$  *reaches*  $i_l \in V_l$  if for some path  $p = (i_1, \dots, i_l)$  from  $V_1$  to  $i_l$  all facilities  $i_1, \dots, i_{l-1}$  are *fully paid* and

$$v_j = c_{jp} + \sum_{i \in p} t_{ij}. \quad (3.2)$$

If, in addition, also  $i_l$  is fully paid, we say that  $j$  *leaves*  $i_l$  or, in case  $l = k$ , that  $j$  gets *connected* (along  $p$  to  $i_k \in V_k$ ).

Our algorithm for constructing the dual solution is a dual ascent method, generalizing the approach in [JV99]. We start with  $v \equiv t \equiv 0$  and increase all  $v_j$  uniformly ( "with unit speed" ). When some  $j \in D$  reaches a not fully paid node  $i \in F$ , we start increasing  $t_{ij}$  with unit speed, until  $f_i$  is fully paid and  $j$  leaves  $i$ . We stop increasing  $v_j$  when  $j$  gets connected. The algorithm maintains the invariant that at time  $T$  the dual variables  $v_j$  that are still being raised are all equal to  $T$ . More precisely, we proceed as described below.

UNTIL all  $j \in D$  are connected DO

- Increase  $v_j$  for all  $j \in D$  not yet connected
- Increase  $t_{ij}$  for all  $i \in F, j \in D$  satisfying (i) – (iii):
  - (i)  $j$  has reached  $i$
  - (ii)  $j$  is not yet connected
  - (iii)  $i$  is not yet fully paid.

Let  $(v, t)$  denote the final dual solution. Before constructing a corresponding primal solution  $(x, y)$ , let us state a few simple facts about  $(v, t)$ .

For each fully paid facility  $i \in V_l, l \geq 2$ , denote by  $T_i$  the time when facility  $i$  became fully paid. *The predecessor* of  $i$  will be the facility in the level  $l - 1$  via which  $i$  was for the first time reached by a demand point, i.e.,

$$Pred(i) = \left\{ i' \in V_{l-1} \mid i' \text{ is fully paid and } T_{i'} + c_{i'i} = \min_{\substack{i'' \in V_{l-1} \\ i'' \text{ fully paid}}} (T_{i''} + c_{i''i}) \right\}.$$

(Ties are broken arbitrarily.)

*The predecessor* of a fully paid facility  $i \in V_1$  will be its closest demand point. We can define the time  $T_{Pred(i)} = 0$ .

For all fully paid facilities  $i$  in the  $k$ -th level denote by  $j_i, p_i = (i_1, \dots, i_k)$  the path through the following points:

- $i_k = i$
- $i_l = Pred(i_{l+1})$ , for each  $1 \leq l \leq k - 1$
- $j_i = Pred(i_1)$ .

We will call the *neighborhood* of  $i$  the set of demand nodes *contributing* to  $p_i$  i.e.,

$$N_i = \{j \in D \mid t_{i'j} > 0 \text{ for some } i' \in p_i\}.$$

Since each  $j \in D$  gets connected we may fix for each  $j \in D$  a connecting path  $\tilde{p}_j \in P$  of fully paid facilities (ties are broken arbitrarily).

LEMMA 3.1.

- (i)  $c(j\tilde{p}_j) \leq v_j$  for all  $j \in D$
- (ii) For all  $j \in D$  and  $i \in V_k$  fully paid such that  $i \in \tilde{p}_j$ , either  $v_j = T_i$  and  $t_{ij} > 0$  or  $v_j > T_i$  and  $t_{ij} = 0$
- (iii) For all fully paid facilities  $i \in V_k$  and corresponding paths  $p_i = (i_1, \dots, i_k)$ , the following relation holds

$$T_{i_1} \leq \dots \leq T_{i_k}$$

- (iv) Let  $i \in V_k$  be a fully paid facility and  $p_i = (i_1, \dots, i_k)$  its associated path. For all  $j \in D$  and  $i_l \in p_i$  with  $t_{i_l j} > 0$ , there exists a path  $p$  from  $V_1$  to  $i_l$  such that

$$c(jp) + \sum_{s=l}^{k-1} c_{i_s i_{s+1}} \leq T_i.$$

In particular,  $c(j_i p_i) \leq T_i$

- (v) If  $i, i'$  are two fully paid facilities in  $V_k$  with intersecting neighborhoods then for each  $j' \in D$ , such that  $i' \in \tilde{p}_{j'}$ ,  $c_{j_i j'} \leq 4 \max\{T_i, v_{j'}\}$

- (vi)  $\sum_{i' \in p_i} t_{i' j} \leq v_j$  for all  $j \in D$

**Proof**

The first claim is straightforward from (3.2) and the definition of  $\tilde{p}_j$ .

The second claim is based on the observation that at time  $T$  all the  $v$ -values that can be increased are equal with  $T$  and that the final  $v$ -values reflect the times when the demand points get connected. There are two possibilities that a fully paid facility  $i \in V_k$  is on the connecting path of a demand point  $j$ . One is that  $j$  reached  $i$  before  $T_i$  and got connected when  $i$  became fully paid. In this case  $t_{ij} > 0$  and  $v_j = T_i$ . The other possibility is that  $j$  reached  $i$  after  $i$  was fully paid, which means that  $t_{ij} = 0$  and  $v_j > T_i$ .

The definition of a predecessor implies that for each fully paid  $i \in F$

$$c_{Pred(i)i} + T_{Pred(i)} \leq T_i. \tag{3.3}$$

The third claim follows immediately .

For the fourth claim, by adding the inequalities (3.3) for  $i_{l+1}, \dots, i_{k-1}$  one obtains

$$\sum_{s=l}^{k-1} c_{i_s i_{s+1}} + T_{i_l} \leq T_{i_k}.$$

Since  $t_{i_l j} > 0$ , there is a path  $p$  along which  $j$  reached  $i_l$  before  $T_{i_l}$ . Clearly,  $c(jp) \leq T_{i_l}$ , which implies (iv).

For proving (v), let  $j \in N_i \cap N_{i'}$ . Since  $j \in N_i$ , there is an  $i_l \in p_i$  such that  $t_{i_l j} > 0$ . Then by (iv), there exists a path  $q$  from  $V_1$  to  $i_l$  such that  $c(jq) \leq T_{i_l}$ .

Suppose  $p_{i'} = (i'_1, \dots, i'_k)$ . Similarly, there is an  $i'_r \in p_{i'}$  and a path  $q'$  from  $V_1$  to  $i'_r$  such that  $c(jq') + \sum_{s=r}^{k-1} c_{i'_s i'_{s+1}} \leq T_{i'}$ .

Using the triangle inequality and (ii), we obtain

$$\begin{aligned} c_{j_i j'} &\leq c(j_i p_i) + c(jq) + c(jq') + \sum_{s=r}^{k-1} c_{i'_s i'_{s+1}} + c(j' \tilde{p}_{j'}) \\ &\leq 2T_i + T_{i'} + v_{j'} \\ &\leq 2T_i + 2v_{j'} \\ &\leq 4 \max\{T_i, v_{j'}\}. \end{aligned}$$

Finally, for proving the statement in the last claim is enough to show that no demand point  $j$  could increase simultaneously two values  $t_{i_l j}, t_{i_s j}$ , for  $i_l \neq i_s$  and  $i_l, i_s \in p_i$ . This follows from the definition of  $p_i$ , which implies that whenever a demand point reaches a facility on  $p_i$ , the predecessor of that facility should have been already paid, and subsequently all the facilities of  $p_i$  situated on inferior levels. ■

We now describe how to construct a corresponding primal solution  $(x, y)$ . Suppose there are  $r$  fully paid facilities in the last level. Order them according to nondecreasing  $T$ -values, say

$$T_1 \leq \dots \leq T_r.$$

Construct greedily a set  $C \subseteq V_k$  of *centers* which have pairwise disjoint neighborhoods and assign each  $j \in D$  to some center  $i_0 \in C$  as follows:

```

INITIALIZE  $C = \emptyset$ 
FOR  $i = 1, \dots, r$  DO
  IF  $N_i \cap N_{i_0} \neq \emptyset$  for some  $i_0 \leq i$ , assign to  $p_{i_0}$  all demand nodes
   $j \in D$  with  $i \in \tilde{p}_j$ 
  ELSE  $C = C \cup \{i\}$  and assign to  $p_i$  all the demand nodes  $j \in D$ 
  with the property that  $i \in \tilde{p}_j$ 

```

The paths  $p_i$  ( $i \in C$ ) are called *central paths*.

**Remark** Note that each demand point  $j$  is assigned to one center. Furthermore, by construction of  $C$ ,  $j$  "contributes" to at most one central path (not necessarily the one to which it is assigned).

The primal solution  $(x, y)$  is obtained by connecting all demand nodes along their corresponding central paths:

$$x_{jp} := \begin{cases} 1 & \text{if } p = p_i \text{ and } j \text{ was assigned to } p_i \\ 0 & \text{otherwise} \end{cases}$$

and

$$y_i := \begin{cases} 1 & \text{if } i \text{ is on a central path} \\ 0 & \text{otherwise} \end{cases}.$$

The shipping cost  $c(x)$  is easily bounded as follows.

If  $j \in D$  is assigned to  $p_{i_0}$  then  $T_{i_0} \leq T_i$ , where  $\{i\} = \tilde{p}_j \cap V_k$ . Due to Lemma 3.1 (ii) and (v), we get  $T_{i_0} \leq v_j$  and

$$c_{jp_{i_0}} \leq c_{j_{i_0}j} + c_{j_{i_0}p_{i_0}} \leq 4v_j + T_{i_0} \leq 5v_j.$$

The cost of opening facilities along a central path  $p_{i_0}$  can be also bounded with the help of Lemma 3.1(vi)

$$\sum_{i \in p_{i_0}} f_i = \sum_{i \in p_{i_0}} \sum_{j \in N_i} t_{ij} \leq \sum_{j \in N_{i_0}} v_j.$$

Since the centers have pairwise disjoint neighborhoods, we further conclude that

$$f(y) = \sum_{i_0 \in C} \sum_{i \in p_{i_0}} f_i \leq \sum_{j \in D} v_j.$$

We have proved

**THEOREM 3.1** *The above primal solution  $(x, y)$  satisfies*

$$c(x) + f(y) \leq 6 \sum_{j \in D} \nu_j.$$

### 3.1 A capacitated version

The following capacitated version has been considered in the literature (cf. [GMM00]): Each  $i \in F$  has an associated *node capacity*  $u_i \in N$  which is an upper bound on the number of paths using  $i$ . On the other hand, we are allowed to open as many copies of  $i$  (at cost  $f_i$  each) as needed.

To formulate this, we replace the 0–1 variables  $y_i$  in  $(P_{int})$  by nonnegative integer variables  $y_i \in Z_+$ , indicating the number of open copies of  $i \in F$ . Furthermore, we add *capacity constraints*

$$\sum_{j \in D} \sum_{p \ni i} x_{jp} \leq u_i y_i, \text{ for each } i \in F. \quad (4.1)$$

Again, we let  $C_{LP}$  denote the optimum value of the corresponding  $LP$ -relaxation.

The idea to approach the capacitated case (also implicit in [JV99] for the 1-level case) is to move the capacity constraints to the objective using Lagrangian multipliers  $\lambda_i \geq 0$ , for each  $i \in F$ . This results in an uncapacitated problem

$$\begin{aligned} C(\lambda) &:= \text{minimize } c(x) + f(y) + \sum_{i \in F} \lambda_i \left( \sum_{j \in D} \sum_{p \ni i} x_{jp} - u_i y_i \right) \\ &= \text{minimize } \tilde{c}(x) + \tilde{f}(y) \end{aligned}$$

with  $\tilde{f}_i = f_i - \lambda_i u_i$ , for each  $i \in F$  and  $\tilde{c}(e) = c(e) + \lambda_i$  if  $i$  is the endpoint of  $e \in E$ . Note that each  $\lambda \geq 0$  gives  $C(\lambda) \leq C_{LP}$ .

As in section 3, we compute a primal 0–1 solution  $(x, y)$  of  $C(\lambda)$  with

$$\tilde{c}(x) + \tilde{f}(y) \leq 6C(\lambda).$$

Note that this does not necessarily satisfy the capacity constraints (4.1). However, a clever choice of the Lagrangian multipliers  $\lambda_i = \frac{1}{2} \frac{f_i}{u_i}$  ( $i \in F$ ) yields

$$\begin{aligned} \tilde{c}(x) + \tilde{f}(y) &= c(x) + \frac{1}{2} \sum_{i \in F} \frac{f_i}{u_i} \sum_{p \ni i} \sum_{j \in D} x_{jp} + \frac{1}{2} \sum_{i \in F} f_i y_i \\ &\geq c(x) + \frac{1}{2} \sum_{i \in F} f_i \bar{y}_i, \end{aligned}$$

where  $\bar{y}_i := \left\lceil \frac{1}{u_i} \sum_{p \ni i} \sum_{j \in D} x_{jp} \right\rceil$  opens each facility  $i \in F$  sufficiently many times. Hence  $(x, \bar{y})$  is indeed a feasible solution of the capacitated problem satisfying

$$c(x) + \frac{1}{2} f(\bar{y}) \leq 6C(\lambda) \leq 6C_{LP},$$



hence

$$c(x) + f(\bar{y}) \leq 12C_{LP}.$$

**Theorem 4.1** Our greedy dual ascent method yields a 12–approximation of the multilevel capacitated facility location problem.

## References

- [ A96] K. Aardal On the two-level uncapacitated facility location problem. *INFORMS Journal on Computing*, 8, pages 289-301, 1996
- [ ACS99] K.I.Aardal, F.Chudak and D.B.Shmoys. A 3-approximation algorithm for the k-level uncapacitated facility location problem. *Information Processing Letters*, 72, pages 161-167, 1999
- [ C98] F. Chudak. Improved approximation algorithms for the uncapacitated facility location. In: R.E. Bixby, E.A. Boyd, R.Z. Rios-Mercado (eds.), *Integer Programming and Combinatorial Optimization*, 6th International IPCO Conference. *Lecture Notes in Computer Science* 1412, pages 180-194, Springer-Verlag, berlin Heidelberg, 1998
- [ CNW90] G. Cornuejols, G. L. Nemhauser and L. A. Wolsey. The uncapacitated facility location problem. In P. Mirchandani and R. Francis, editors, *Discrete Location Theory*, pages 119-171. John Wiley and Sons, New York, 1990
- [ GK98] S. Guha and S. Khuller. Greedy strikes back: improved facility algorithms. *Proceedings of the 9th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 649-657, 1998
- [ KPR98] M. R. Korupolu, C. G. Plaxton and R. Rajaraman. Analysis of a local search heuristic for facility location problems. *Technical Report 98-30, DIMACS, June 1998*
- [ JV99] K.Jain and V.V.Vazirani. Primal-dual approximation algorithms for metric facility location and  $k$ –median problems. *Proceedings of the Twenty-Ninth Annual IEEE Symposium on Foundations of Computer Science*, 1999
- [ LV92] J. H. Lin and J. S. Vitter.  $\epsilon$ –approximation with minimum packing constraint violation. *Proceedings of the 24th Annual ACM Symposium on Theory of Computing*, pages 771-782, 1992
- [ MMP] A. Meyerson, K. Munagala, S. Plotkin. Cost distance: Two metric network design. *Proceedings of the 41-th IEEE Symposium on Foundation of Computer Science*, 2000
- [ STA97] D. B. Shmoys, E. Tardos and K. I. Aardal. Approximation algorithms for facility location problems. *Proceedings of the 29th Annual ACM Symposium on Theory of Computing*, pages 265-274, 1997
- [ CG99] M. Charikar and S. Guha. Improved combinatorial algorithms for facility location and  $k$ -median problems. *Proceedings of the Twenty-Ninth Annual IEEE Symposium on Foundations of Computer Science*, 1999
- [ GMM00] S. Guha, A. Meyerson, K. Munagala. Hierarchical Placement and Network Design Problems. to appear in *Proceedings of the Thirtyth Annual IEEE Symposium on Foundations of Computer Science, 2000*
- [ CS99] F. Chudak and D. Shmoys. Improved approximation algorithms for the capacitated facility location problem. *Proceedings of the tenth ACM-SIAM Symposium on Discrete Algorithms, 1999*
- [ S97] M. Sviridenko, Personal communication, 1997