A simple dual ascent algorithm for the multilevel facility location problem

A.F. Bumb, W.Kern

Faculty of Mathematical Sciences, University of Twente, P.O. Box 217, 7500 AE Enschede, The Netherlands,
email: a.f.bumb@math.utwente.nl,kern@math.utwente.nl

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 techinique.

1 Introduction

This paper concernes with the multilevel facility location problem. Being an extension of the uncapacitated facility location problem, which is known to be Max SNP-hard (see[2]), this problem is Max SNP-hard as well. The first approximation algorithms for the multilevel facility location problem were developed by Shmoys, Tardos & Aardal [5] and Aardal, Chudak & Shmoys [1] and were based on rounding of an LP solution to an integer one. The performance guarantee of these algorithms were 3.16, respectively 3. The first combinatorial algorithm for the multilevel facility location problem was developed by Meyerson, Munagala&Plotkin [4], and finds a solution within $O(\log |D|)$ the optimum, where D is the set of demand points. Using an idea from [3], 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 extends to a capacitated variant of the problem, when each facility can serve only a certain number of demand points, with an increase of the performance guarantee to 12.

Preprint submitted to Elsevier Science *26 April 2001*

2 The metric multilevel uncapacitated facility location problem

Consider a complete $(k+1)-$partite graph $G=(V,E)$ with $V=V_0 \cup ... \cup V_k$ and $E = \bigcup_{l=1}^{k} V_{l-1} \times V_l$. The set $D = V_0$ is the set of *demand nodes* and $F = V_1 \cup ... \cup V_k$ 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. 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 $p = (v_1,...,v_k) \in P$, we let jp denote the path $(j, v_1, ..., v_k)$. 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 $\sum_{j \in D} c(jp_j) + f(\bigcup_{j \in D} p_j)$. 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

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

Denote with (P) the $LP-$relaxation of the above integer program obtained by allowing (x,y) to be positive reals.

3 The dual ascent algorithm

Introducing dual variables v_j and t_{ij} corresponding to constraints (1) and (2) in (P), the dual of (P) can be written as

$$\text{maximize } \sum_{j \in D} v_j$$

$$\text{subject to} \quad v_j - \sum_{i \in p} t_{ij} \le c(jp), \quad \text{for each } p \in P, \; j \in D \qquad (3)$$

$$\sum_{j \in D} t_{ij} \le f_i, \qquad \text{for each } i \in F \qquad (4)$$

$$t_{ij} \ge 0, \qquad \text{for each } i \in F, \; j \in D$$

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 $\sum_{j \in D} t_{ij} = f_i$. A demand point $j \in D$ *reaches* $i_l \in V_l$ if for some path $p = (i_1, ..., i_l)$ from $V_1$to i_l all facilities $i_1, ...i_{l-1}$ are *fully paid* and $v_j = c_{jp} + \sum_{i \in p} t_{ij}$. 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 [3].
We start with $v \equiv t \equiv 0$ and increase all v_j uniformly ("with unit speed"). Until all demand points get connected we proceed as follows .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.
Let (v, t) denote the final dual solution. For each fully paid facility $i \in V_l$, $l \ge 2$,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.,

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

(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 $p_i = (i_1, ...i_k)$ the path through the following points:

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

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 $\widetilde{p_j} \in P$ of fully paid facilities (ties are broken arbitrarily). 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 $T_1 \le ...T_r$. Construct greedily a set $C \subseteq V_k$ of *centers* which have parewise disjoint neighborhoods and assign each $j \in D$

to some center $i_0 \in C$ in the following way: For each fully paid facility i in the last level check if there is a center $i_0 \leq i$ such that $N_i \cap N_{i_0} \neq \emptyset$. If such an i_0 exist, assign to p_{i_0} all demand nodes $j \in D$ with $i \in \widetilde{p_j}$. Otherwise add i to C and assign to p_i all the demand nodes $j \in D$ with $i \in \widetilde{p_j}$.
The primal solution (x, y) is obtained as follows:

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

and

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

Theorem 1 *The above primal solution (x, y) satisfies*

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

implying that the performance guarantee of algorithm is 6.

NOTE: The greedy dual ascent algorithm combined with a technique based on Lagrangian multipliers yields a 12 approximation algorithm for the capacitated variant of the multilevel facility location problem in which each facility can serve only a certain number of demand points.

References

[1] 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

[2] 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

[3] 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

[4] 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

[5] 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*