Primal-Dual Approximation Algorithms for Metric Facility Location and k-Median Problems

Kamal Jain * *Vijay V. Vazirani* *

Abstract

We present approximation algorithms for the metric uncapacitated facility location problem and the metric k-median problem achieving guarantees of 3 and 6 respectively. The distinguishing feature of our algorithms is their low running time: $O(m \log m)$ and $O(m \log m(L+\log(n)))$ respectively, where n and m are the total number of vertices and edges in the underlying graph. The main algorithmic idea is a new extension of the primal-dual schema.

1. Introduction

Given costs for opening facilities and costs for connecting cities to facilities, the uncapacitated facility location problem seeks a minimum cost solution that connects each city to an open facility. Clearly, this problem is applicable to a number of industrial situations. For this reason it has occupied a central place in operations research since the early 60's [3, 25, 33, 34], and has been studied from the perspectives of worst case analysis, probabilistic analysis, polyhedral combinatorics and empirical heuristics (see [10, 28, 29]). In the last few years, there has been renewed interest in tackling this problem, this time from the perspective of approximation algorithms [12, 23, 24, 27, 32]. In this paper, we carry this further by developing an approximation algorithm based on the primal-dual schema. We further use this algorithm as a subroutine to solve a related problem, the k-median problem. The latter problem differs in that there are no costs for opening facilities, instead a number k is specified, which is an upper bound on the number of facilities that can be opened. The two algorithms achieve approximation guarantees of 3 and 6 respectively.

Both of our algorithms work only for the metric case, i.e., when the connecting costs satisfy the triangle inequality; both problems are **NP**-hard for this case as well. If the connection costs are unrestricted, approximating either problem is as hard as approximating set cover, and therefore cannot be done better than $O(\log n)$ factor, unless **NP** $\subseteq$ $\tilde{\textbf{P}}$. For the first problem, this is straightforward to see, and for the second, this is established by Lin and Vitter [26].

The distinguishing feature of our algorithms is their low running time: $O(m \log m)$ and $O(m \log m(L + \log(n)))$ respectively, where n and m are the total number of vertices and edges in the underlying graph ($n = n_c+n_f$ and $m = n_c \times n_f$, where n_c and n_f are the number of cities and facilities) and L is the number of bits needed to represent a connecting cost. In particular, the running time of the first algorithm is dominated by the time taken to sort the connecting costs of edges. It is worth pointing out that our facility location algorithm is also suitable for distributed computation.

The first constant factor algorithm for the metric uncapacitated facility location problem was given by Shmoys, Tardos and Aardal [32], improving on Hochbaum's bound of $O(\log n)$ [20] (see [27] for another $O(\log n)$ factor algorithm). Their approximation guarantee was 3.16. After some improvements [23, 11], the current best factor is $(1 + 2/e)$, due to Chudak and Shmoys [12]. The drawback of these algorithms, based on LP-rounding, is that they need to solve large linear programs, and so have prohibitive running times for most applications. A different approach was recently used by Korupolu, Plaxton and Rajaraman [24] (see also [14]). They showed that a well known local search heuristic achieves an approximation guarantee of $(5+\epsilon)$, for any $\epsilon > 0$. However, even this algorithm has a high running time of $(n^6 \log n/\epsilon)$. Regarding hardness results, the work of [19, 35] establishes that a better factor than 1.463 is not possible, unless **NP** $\subseteq$ $\tilde{\textbf{P}}$.

Researchers have felt that the primal-dual schema should be adaptable in interesting ways to the combinatorial structure of individual problems, and that its full potential has not yet been realized in the area of approximation algorithms. Our work substantiates this belief. We extend the scope of this schema in the following way: All primal-dual approximation algorithms obtained so far [6, 18, 36, 17, 31, 30, 21] work with a pair of covering and packing linear programs, i.e., a primal-dual pair of LP's such that all components of the constraint matrix, objective function vector and right hand side vector are non-negative. This includes, for instance, [36, 17]

*Address: College of Computing, Georgia Institute of Technology, Atlanta, GA 30332–0280, E-mail: {kjain,vazirani}@cc.gatech.edu. Research supported by NSF Grant CCR-9820896.

in which the overall LP-relaxation does have negative coefficients; however, the problem is decomposed into phases, and the relaxation used in each phase is a covering program. On the other hand, our algorithm works with primal and dual programs that do have negative coefficients.

Despite this added complexity, our algorithm has a particularly simple description: Each city j keeps raising its dual variable, α_j, until it gets connected to an open facility. All other primal and dual variables simply respond to this change, trying to maintain feasibility or satisfying complementary slackness conditions. For the latter, we give a new mechanism as well.

Until the work of [30] (which relaxed the dual program itself), all approximation algorithms based on the primal-dual schema used the mechanism formalized in [36]: In the first phase, an integral primal solution is found, satisfying the primal complementary slackness conditions; however, this solution may have redundancies. In the second phase, a minimal solution is extracted, typically via a reverse delete procedure, and in the process, dual complementary slackness conditions get satisfied with a relaxation factor. The final algorithm has this factor as its approximation guarantee.

Our first phase is similar. For the second phase, we introduce the new procedure of *forward include* for removing redundancies. After this procedure is done, *all* complementary slackness conditions are satisfied; however, the primal solution may be infeasible. The solution is augmented – this time the primal conditions need to be relaxed by a factor of 3, which is also the approximation guarantee of the algorithm.

The k-median problem also has numerous applications, especially in the context of clustering, and has also been extensively studied. In recent years, ths problem has found new clustering applications in the area of data mining (see [7]).

A non-trivial approximation algorithm for this problem eluded researchers for many years. The breakthrough was made by Bartal, who gave a factor $O(\log n \log\log n)$ algorithm. After an improvement [8], a constant factor algorithm, using a different approach, was recently obtained by Charikar, Guha, Tardos and Shmoys [9]. Their algorithm has an approximation guarantee of $6\frac{2}{3}$; however, it has the same drawback since it uses LP-rounding. Their algorithm uses several ideas from the constant factor algorithms obtained for the facility location problem, thus making one wonder if there is a deeper connection between the two problems.

In this paper, we establish such a connection: between the LP-relaxations for the two problems. This enables us to use our algorithm for the facility location problem as a subroutine to solve the k-median problem. The idea for this lies in the following principle from economics: taxation is an effective way of controlling the amount of goods coming across the border – raising tariffs will reduce in-flow and vice versa. Given an instance of the k-median problem, we remove the restriction that at most k facilities be opened, and instead assign a cost of z for opening each of the facilities, thus obtaining an instance of the facility location problem. By changing z, we can control the number of facilities opened by our facility location algorithm. Ideally, at this point, we would like to find a value of z for which the algorithm opens exactly k facilities. We do not know how to do this. Instead, we find two solutions for "close" values of z, one opening more than k facilities, and the other opening less. An appropriate convex combination of these solutions is found that opens, fractionally, exactly k facilities. Finally, using a randomized rounding procedure, this is converted into an integral solution, sacrificing a small multiplicative factor in the process. A derandomization of this procedure is also provided.

These ideas also help solve a common generalization of the two problems – in which facilities have costs, and in addition, there is an upper bound on the number of facilities that can be opened. We give a factor 6 approximation algorithm for this problem as well; the previous bound was 9.8 [9].

The *capacitated* facility location problem, in which each facility i can serve at most u_i cities, has no non-trivial approximation algorithms. Part of the problem is that all LP-relaxations known for this problem have unbounded integrality gap (see [32]). In Section 5 we give a factor 4 approximation algorithm for the variant in which each facility can be opened an unbounded number of times; if facility i is opened y_i times, it can serve at most $u_i y_i$ cities. A special case of this version, in which the capacities of all the facilities are assumed to be equal, is solved with factor 3 in [13], again using LP-rounding.

2. The metric uncapacitated facility location problem

The *uncapacitated facility location problem* seeks a minimum cost way of connecting cities to open facilities. It can be stated formally as follows: Let G be a bipartite graph with bipartition (F, C), where F is the set of *facilities* and C is the set of *cities*. Let f_i be the cost of opening facility i, and c_{ij} be the cost of connecting city j to (opened) facility i. The problem is to find a subset $I \subseteq F$ of facilities that should be opened, and a function $\phi : C \rightarrow I$ assigning cities to open facilities in such a way that the total cost of opening facilities and connecting cities to open facilities is minimized. We will consider the *metric* version of this problem, i.e., the c_{ij}'s satisfy the triangle inequality.

We will adopt the following notation: $|C| = n_c$ and $|F| = n_f$. The total number of vertices $n_c + n_f = n$ and the total number of edges $n_c \times n_f = m$.

Following is an integer program for this problem. In this program, y_i is an indicator variable denoting whether facility i is open, and x_{ij} is an indicator variable denoting whether city j is connected to the facility i. The first constraint ensures that each city is connected to at least one facility, and the second ensures that this facility must be open.

$$\begin{aligned} \text{minimize} \quad & \sum_{i \in F, j \in C} c_{ij} x_{ij} + \sum_{i \in F} f_i y_i \\ \text{subject to} \quad & \forall j \in C: \sum_{i \in F} x_{ij} \geq 1 \\ & \forall i \in F, j \in C: \ y_i - x_{ij} \geq 0 \\ & \forall i \in F, j \in C: \ x_{ij} \in \{0, 1\} \\ & \forall i \in F: \ y_i \in \{0, 1\} \end{aligned} \tag{1}$$

The LP-relaxation of this program is:

$$\begin{aligned} \text{minimize} \quad & \sum_{i \in F, j \in C} c_{ij} x_{ij} + \sum_{i \in F} f_i y_i \\ \text{subject to} \quad & \forall j \in C: \sum_{i \in F} x_{ij} \geq 1 \\ & \forall i \in F, j \in C: \ y_i - x_{ij} \geq 0 \\ & \forall i \in F, j \in C: \ x_{ij} \geq 0 \\ & \forall i \in F: \ y_i \geq 0 \end{aligned} \tag{2}$$

The dual program is:

$$\begin{aligned} \text{maximize} \quad & \sum_{j \in C} \alpha_j \\ \text{subject to} \quad & \forall i \in F, j \in C: \ \alpha_j - \beta_{ij} \leq c_{ij} \\ & \forall i \in F: \sum_{j \in C} \beta_{ij} \leq f_i \\ & \forall j \in C: \ \alpha_j \geq 0 \\ & \forall i \in F, j \in C: \ \beta_{ij} \geq 0 \end{aligned} \tag{3}$$

2.1. Relaxing primal complementary slackness conditions

Our algorithm is based on the primal-dual schema. As stated in the introduction, instead of the usual mechanism of relaxing dual complementary slackness conditions, we relax the primal conditions. Before showing how this is done, let us give the reader some feel for how the dual variables "pay" for a primal solution by considering the following simple setting: suppose LP (2) has an optimal solution that is integral, say $I \subseteq F$ and $\phi : C \rightarrow I$. Thus, under this solution, $y_i = 1$ iff $i \in I$, and $x_{ij} = 1$ iff $i = \phi(j)$.

Let $(\boldsymbol{\alpha}, \boldsymbol{\beta})$ denote an optimal dual solution. The reader can verify that primal and dual complementary slackness conditions imply the following facts:

- Each open facility is fully paid for, i.e., if $i \in I$, then
$$\sum_{j:\ \phi(j)=i} \beta_{ij} = f_i.$$
- Suppose city j is connected to facility i, i.e., $\phi(j) = i$. Then, j does not contribute for opening any facility besides i, i.e., $\beta_{i'j} = 0$ if $i' \neq i$. Furthermore, $\alpha_j - \beta_{ij} = c_{ij}$. So, we can think of α_j as the total price paid by city j; of this, c_{ij} goes towards the use of edge (i, j), and β_{ij} is the contribution of j towards opening facility i.

Suppose the primal complementary slackness conditions were relaxed as follows, while maintaining the dual conditions:

$$\forall j \in C: \ (1/3)c_{\phi(j)j} \ \leq \ \alpha_j - \beta_{\phi(j)j} \leq c_{\phi(j)j},$$

and

$$\forall i \in I: \ (1/3)f_i \ \leq \sum_{j:\ \phi(j)=i} \beta_{ij} \ \leq \ f_i.$$

Then, the cost of the (integral) solution found would be within thrice the dual found, thus leading to a factor 3 approximation algorithm. However, we would like to obtain the stronger inequality stated in Theorem 7, so as to use this algorithm to solve the k-median problem. For this reason, we will relax the primal conditions as follows: The cities are partitioned into two sets, *directly connected* and *indirectly connected*.

Only directly connected cities will pay for opening facilities, i.e., β_{ij} can be non-zero only if j is a directly connected city and $i = \phi(j)$. For an indirectly connected city j, the primal condition is relaxed as follows:

$$(1/3)c_{\phi(j)j} \leq \alpha_j \leq c_{\phi(j)j}.$$

All other primal conditions are maintained, i.e., for a directly connected city j,

$$\alpha_j - \beta_{\phi(j)j} = c_{\phi(j)j},$$

and for each open facility i,

$$\sum_{j:\ \phi(j)=i} \beta_{ij} = f_i.$$

2.2. The algorithm

Our algorithm consists of two phases. In Phase 1, the algorithm operates in a primal-dual fashion. It finds a dual feasible solution, and also determines a set of tight edges and temporarily open facilities, F_t. Phase 2 consists of a forward include step which chooses a subset I of F_t to open. A mapping, ϕ, from cities to I is also determined.

Algorithm 1

Phase 1
The various steps executed in this phase derive from the following underlying process: Each city j keeps raising its dual variable, α_j, until it gets connected to an open facility. All other primal and dual variables simply respond to this change, trying to maintain feasibility or satisfying complementary slackness conditions.

A notion of *time* is defined in this phase, so that each event can be associated with the time at which it happened; the phase starts at time 0. Initially, each city is defined to be *unconnected*. Throughout this phase, the algorithm raises the dual variable α_j for each unconnected city j uniformly at unit rate, i.e., α_j will grow by 1 in unit time. When $\alpha_j = c_{ij}$ for some edge (i, j), the algorithm will declare this edge to be *tight*. At this point, the dual variable β_{ij} is raised uniformly, thus ensuring that the first constraint in LP (3) is not violated. β_{ij} goes towards paying for facility i.

Facility i is said to be *paid for* if $\sum_j \beta_{ij} = f_i$. When this happens for a facility i, the algorithm checks whether there is a city j having a tight edge to i such that j is still unconnected. If so, the algorithm declares this facility *temporarily open*. Furthermore, all unconnected cities having tight edges to this facility are declared *connected* and facility i is declared the *connecting witness* for each of these cities. (Notice that the dual variables α_j of these cities are not raised anymore.) In the future, as soon as an unconnected city j gets a tight edge to i, j will also be declared connected and i will be declared to be the connecting witness for j. When all cities are connected, the first phase terminates.

If several events happen simultaneously, the algorithm picks one of them arbitrarily, declares this event, and takes all action needed with this declaration.

Remark 2 The purpose of the last rule is to deal with the following situation: two edges (i_1, j) and (i_2, j) from unconnected city j go tight simultaneously to temporarily open facilities i_1 and i_2. In this case, we want only one of these edges to be declared tight.

Phase 2
Let F_t denote the set of temporarily open facilities and T denote the subgraph of G consisting of edges that were tight at the end of Phase 1. Let T^2 denote the graph that has edge (u, v) iff there is a path of length at most 2 between u and v in T, and let H be the subgraph of T^2 induced on F_t. Now, facilities in F_t are considered in the order in which they were temporarily opened, and a maximal independent subset, I, of these vertices w.r.t. the graph H is picked as follows: While considering facility i, if no neighbor of i, w.r.t. H, is already picked, then i is picked. Otherwise, one of the picked neighbors of i is declared to be the *closing witness* for i, and i is not picked. All facilities in the set I are declared *open*.

Finally, for each city j, if there is a tight edge (i, j) and facility i is open, then $\phi(j) = i$, and city j is declared *directly connected*. Otherwise, consider tight edge (i, j) such that i was the connecting witness for j. Since $i \notin I$, its closing witness, say i', must be open. Let $\phi(j) = i'$ and city j is declared *indirectly connected*.

I and ϕ define a primal integral solution: $x_{ij} = 1$ iff $\phi(j) = i$, and $y_i = 1$ iff $i \in I$. The values of α_j and β_{ij} obtained at the end of Phase 1 form a dual feasible solution.

2.3. Analysis

We will show how the dual variables α_j's pay for the primal costs of opening facilities and connecting cities to facilities. Denote by α_j^f and α_j^e the contributions of city j to these two costs respectively; $\alpha_j = \alpha_j^f + \alpha_j^e$. If j is indirectly connected, then $\alpha_j^f = 0$ and $\alpha_j^e = \alpha_j$. If j is directly connected, then the following must hold:

$$\alpha_j = c_{ij} + \beta_{ij},$$

where $i = \phi(j)$. Now, let $\alpha_j^f = \beta_{ij}$ and $\alpha_j^e = c_{ij}$.

Lemma 3 *Let $i \in I$. If for city j, (i, j) was tight at the end of Phase 1, then $\phi(j) = i$.*

Proof : For city j, define

$$\mathcal{F}_j = \{i \in F_t \mid (i, j) \text{ is tight at the end of Phase 1}\}.$$

In H, $\mathcal{F}_j$ forms a clique, and so at most one of these facilities is picked in I. In case one of these facilities, say i is picked, j will be declared directly connected to i. Viewing from the perspective of $i \in I$, all cities having tight edges to i must be directly connected to it. (Of course, there may be addition cities that are indirectly connected to i.) □

Lemma 4 *Let $i \in I$. Then,*

$$\sum_{j:\, \phi(j)=i} \alpha_j^f = f_i.$$

Proof : The condition for declaring facility i temporarily open during Phase 1 was that

$$\sum_{j:\, (i,j) \text{ is tight}} \beta_{ij} = f_i.$$

After i is declared temporarily open, β_{ij} remains unchanged for each j. Therefore, the condition given above holds even at the end of Phase 1. By Lemma 3 and the definition of α_j^f, the lemma follows (notice that if $\phi(i) = j$, but (i, j) is not tight, then $\alpha_j^f = 0$). □

Corollary 5 $\sum_{i \in I} f_i = \sum_{j \in C} \alpha_j^f$.

Recall that α_j^f was defined to be 0 for indirectly connected facilities. So, only the directly connected cities pay for the cost of opening facilities.

Lemma 6 *For an indirectly connected city j, $c_{ij} \leq 3\alpha_j^e$, where $i = \phi(j)$.*

Proof : Since j is indirectly connected to i, there must be a tight edge (i', j) such that i was the closing witness for i' and i' was the connecting witness for j. Since (i, i') is an edge in T^2, there must be a city that has tight edges to both i and i'; let j' be any such city. Let t_1 and t_2 be the times at which i and i' were declared temporarily open. Since i is the closing witness for i', $t_1 \leq t_2$.

Since edge (i', j) is tight, $\alpha_j \geq c_{i',j}$. Furthermore, since i' was the the connecting witness for j, $\alpha_j \geq t_2$ (notice that this holds regardless of whether edge (i', j) went tight before or after i' was declared temporarily open).

Finally, we claim that $t_2 \geq c_{ij'}$ and $t_2 \geq c_{i',j'}$. Suppose not. Consider the edge which went tight later, and the instant at which it was declared tight. At this instant, i and i' had both been declared temporarily open, and so j' must have already been declared connected (via the edge that went tight first). This leads to a contradiction since an already connected city cannot get additional tight edges. Notice that the rule for resolving ties is important for this argument; see Remark 2.

Hence we get that the cost of each of the three edges (i', j), (i', j') and (i, j') is bounded by α_j. The lemma follows by the triangle inequality. □

Theorem 7 *The primal and dual solutions constructed by the algorithm satisfy:*

$$\sum_{i \in F,\, j \in C} c_{ij} x_{ij} + 3 \sum_{i \in F} f_i y_i \leq 3 \sum_{j \in C} \alpha_j.$$

Proof : For a directly connected city j, $c_{ij} = \alpha_j^e \leq 3\alpha_j^e$, where $\phi(j) = i$. Combining with Lemma 6 we get

$$\sum_{i \in F, j \in C} c_{ij} x_{ij} \leq 3 \sum_{j \in C} \alpha_j^e.$$

Adding this to the following inequality obtained from Corollary 5 gives the theorem:

$$3 \sum_{i \in I} f_i \leq 3 \sum_{j \in C} \alpha_j^f.$$

□

2.4. Running time

Sort all edges by increasing cost – this gives the order and the times at which edges go tight. For each facility, i, we maintain its unpaid cost and the current number of cities that are contributing towards its cost (i.e., unconnected cities having tight edges to this facility); these are initialized to f_i and 0 respectively. The ratio of these gives the time after which facility i will go tight, assuming no other event happens on the way.

So, each iteration takes $O(n_f)$ time and there are $O(n_c)$ iterations. Therefore, besides the sorting step, the rest of the algorithm takes linear time, i.e., $O(m)$ time. The overall running time of the algorithm is $O(m \log m)$. Hence, we get:

Theorem 8 *Algorithm 1 achieves an approximation factor of 3 for the facility location problem, and has a running time of $O(m \log m)$.*

Remark 9 Notice that changing the way in which ties are resolved in Algorithm 1 cannot change the dual solution found, it can change the primal only.

2.5. Tight example

The following infinite family of examples shows that the analysis of our algorithm is tight: The graph has n cities, $c_1, c_2, \ldots, c_n$ and two facilities f_1 and f_2. Each city is at a distance of 1 from f_2. City c_1 is at a distance of 1 from f_1, and $c_2, \ldots, c_n$ are at a distance of 3 from f_1. The opening cost of f_1 and f_2 are ϵ and $(n+1)\epsilon$ respectively, for a small number ϵ.

The optimal solution is to open f_2 and connect all cities to it, at a total cost of $(n+1)\epsilon + n$. Algorithm 1 will however open facility f_1 and connect all cities to it, at a total cost of $\epsilon + 1 + 3(n-1)$.

2.6. Extension to arbitrary demands

A small extension to Algorithm 1 enables it to handle the following generalization to arbitrary demands: For each city j, a non-negative demand d_j is specified; any open facility can serve this demand. The cost of serving this demand via facility i will be $c_{ij} d_j$.

The only change to IP (1) and LP (2) is that in the objective function, $c_{ij}x_{ij}$ is replaced by $c_{ij}d_jx_{ij}$. This changes the first constraint in the dual (3) to

$$\forall i \in F,\ j \in C:\ \alpha_j - \beta_{ij} \le c_{ij}d_j.$$

The only change to Algorithm 1 is that for each city j, α_j is raised at rate d_j. Notice that because of the change in the first constraint in the dual, edge (i,j) still goes tight at time c_{ij}. However, once (i,j) goes tight, β_{ij} will be increasing at rate d_j, and so facility i may get opened earlier than in the unit demands case.

An easy way to see that this modification works is to reduce to the unit demands case by making d_j copies of city j. (The change proposed above to Algorithm 1 is more general, since it works even if d_j is non-integral, and even if it is exponentially large.)

3. The metric k-median problem

The k-median problem differs from the facility location problem in two respects: there is no cost for opening facilities, and there is an upper bound, k, on the number of facilities that can be opened; k is not fixed, it is supplied as part of the input. Once again, we will assume that the edge costs satisfy the triangle inequality.

Since the two problems are similar, so are their integer programs and LP-relaxations. Our algorithm exploits this similarity in order to reduce the k-median problem to the facility location problem. Following is an integer program for the k-median problem. The indicator variables y_i and x_{ij} play the same role as in (1).

$$\begin{array}{ll} \text{minimize} & \sum_{i\in F, j\in C} c_{ij}x_{ij} \\ \text{subject to} & \forall j \in C:\ \sum_{i\in F} x_{ij} \ge 1 \\ & \forall i \in F, j \in C:\ y_i - x_{ij} \ge 0 \\ & \sum_{i\in F} -y_i \ge -k \\ & \forall i \in F, j \in C:\ x_{ij} \in \{0,1\} \\ & \forall i \in F:\ y_i \in \{0,1\} \end{array} \qquad (4)$$

The LP-relaxation of this program is:

$$\begin{array}{ll} \text{minimize} & \sum_{i\in F, j\in C} c_{ij}x_{ij} \\ \text{subject to} & \forall j \in C:\ \sum_{i\in F} x_{ij} \ge 1 \\ & \forall i \in F, j \in C:\ y_i - x_{ij} \ge 0 \\ & \sum_{i\in F} -y_i \le -k \\ & \forall i \in F, j \in C:\ x_{ij} \ge 0 \\ & \forall i \in F:\ y_i \ge 0 \end{array} \qquad (5)$$

The dual program is:

$$\begin{array}{ll} \text{maximize} & \sum_{j\in C} \alpha_j - zk \\ \text{subject to} & \forall i \in F, j \in C:\ \alpha_j - \beta_{ij} \le c_{ij} \\ & \forall i \in F:\ \sum_{j\in C} \beta_{ij} \le z \\ & \forall j \in C:\ \alpha_j \ge 0 \\ & \forall i \in F, j \in C:\ \beta_{ij} \ge 0 \\ & z \ge 0 \end{array} \qquad (6)$$

3.1. The high level idea

The similarity in the linear programs is exploited as follows: Take an instance of the k-median problem, assign a cost of z for opening each facility, and find optimal solutions to LP (2) and LP (3), say $(\boldsymbol{x}, \boldsymbol{y})$ and $(\boldsymbol{\alpha}, \boldsymbol{\beta})$ respectively. By the strong duality theorem,

$$\sum_{i\in F,\ j\in C} c_{ij}x_{ij} + \sum_{i\in F} zy_i \ = \ \sum_{j\in C} \alpha_j.$$

Now, suppose that the primal solution $(\boldsymbol{x}, \boldsymbol{y})$ happens to open exactly k facilities (fractionally), i.e., $\sum_i y_i = k$. Then, we claim that $(\boldsymbol{x}, \boldsymbol{y})$ and $(\boldsymbol{\alpha}, \boldsymbol{\beta}, z)$ are optimal solutions to LP (5) and LP (6) respectively. Feasibility is easy to check. Optimality follows by substituting $\sum_i y_i = k$ in the above equality, and rearranging terms to show that the primal and dual solutions achieve the same objective function value:

$$\sum_{i\in F,\ j\in C} c_{ij}x_{ij} \ = \ \sum_{j\in C} \alpha_j - zk.$$

Let's use this idea, together with Algorithm 1 and Theorem 7, to obtain a "good" integral solution to LP (5). Suppose with a cost of z for opening each facility, Algorithm 1 happens to find solutions $(\boldsymbol{x}, \boldsymbol{y})$ and $(\boldsymbol{\alpha}, \boldsymbol{\beta})$, where the primal solution opens exactly k facilities. By Theorem 7,

$$\sum_{i\in F,\ j\in C} c_{ij}x_{ij} + 3zk \ \le \ 3\sum_{j\in C} \alpha_j.$$

Now, observe that $(\boldsymbol{x}, \boldsymbol{y})$ and $(\boldsymbol{\alpha}, \boldsymbol{\beta}, z)$ are primal (integral) and dual feasible solutions to the k-median problem satisfying

$$\sum_{i\in F,\ j\in C} c_{ij}x_{ij} \ \le \ 3(\sum_{j\in C} \alpha_j - zk).$$

Therefore, $(\boldsymbol{x}, \boldsymbol{y})$ is a solution to the k-median problem within thrice the optimal.

Notice that proof of factor 3 given above would not work if less than k facilities were opened; if more than k facilities are opened, the solution is infeasible for the k-median problem. The remaining problem is to find a value of z so that *exactly* k facilities are opened. Several ideas are required for this. The first is the following principle from economics: taxation is an effective way of controlling the amount of goods coming across the border – raising tariffs will reduce in-flow and vice versa. In a similar manner, raising z should reduce the number of facilities opened and vice versa.

It is natural now to seek a modification to Algorithm 1 that can find a value of z and a way of resolving ties so that exactly k facilities get opened. This would lead to a factor 3 approximation algorithm. We don't know if this is possible. Instead, we present the following strategy which leads to a factor 6 algorithm. For the rest of the discussion, assume that we never encountered a run of the algorithm which resulted in exactly k facilities being opened.

Clearly, when $z = 0$, the algorithm will open all facilities, and when z is very large, it will open only one facility. We will show in Section 3.2 that there is a value of z and a way of resolving ties appropriately, for which Algorithm 1 will find two solutions, one with $k_1 < k$ facilities and the other with $k_2 > k$ facilities. By Remark 9, the dual solutions found in the two runs will be identical, say $(\boldsymbol{\alpha}, \boldsymbol{\beta})$. Let $(\boldsymbol{x}^s, \boldsymbol{y}^s)$ and $(\boldsymbol{x}^l, \boldsymbol{y}^l)$ be the two primal solutions found, with $\sum_{i \in F} y_i^s = k_1$ and $\sum_{i \in F} y_i^l = k_2$ (the superscripts s and l denote "small" and "large" respectively).

Now, by Theorem 7 we have:

$$\sum_{i \in F,\ j \in C} c_{ij} x_{ij}^s \ \le\ 3(\sum_{j \in C} \alpha_j - z k_1),$$

and

$$\sum_{i \in F,\ j \in C} c_{ij} x_{ij}^l \ \le\ 3(\sum_{j \in C} \alpha_j - z k_2).$$

Let $(\boldsymbol{x}, \boldsymbol{y}) = a(\boldsymbol{x}^s, \boldsymbol{y}^s) + b(\boldsymbol{x}^l, \boldsymbol{y}^l)$ be a convex combination of these two solutions, with $ak_1 + bk_2 = k$; under these conditions, $a = (k - k_1)/(k_2 - k_1)$ and $b = (k_2 - k)/(k_2 - k_1)$. Since $(\boldsymbol{x}, \boldsymbol{y})$ is a feasible (fractional) solution to the facility location problem that opens exactly k facilities, it is also a feasible (fractional) solution to the k-median problem. Furthermore, $(\boldsymbol{\alpha}, \boldsymbol{\beta}, z)$ is a feasible solution to the dual of the k-median problem.

The two inequalities given above yield:

$$\sum_{i \in F,\ j \in C} c_{ij} x_{ij} \ \le 3(\sum_{j \in C} \alpha_j - kz).$$

Therefore, we have obtained a solution to the k-median problem which is within thrice the optimal, and in which each city is (fractionally) serviced by at most two facilities. In Section 3.3 we give a randomized rounding procedure that obtains an integral solution to the k-median problem from $(\boldsymbol{x}, \boldsymbol{y})$, in the process at most doubling the cost. Finally, in Section 3.4 we derandomize this procedure.

3.2. Binary search

Let us fix an arbitrary ordering $\mathcal{O}$ of all edges and facilities, and require that Algorithm 1 follow this ordering in resolving ties. Under ordering $\mathcal{O}$, for a given value of the parameter z, the order in which edges and facilities go tight is fixed (and so is the primal solution found and the number of facilities opened). By *sequence at* θ we mean the ordered list of edges and facilities that go tight for $z = \theta$. Consider changes in this sequence as z changes. Let us say that a value of z is *critical* if an infinitesimal change results in a change in the sequence.

Let θ be a critical value of z, with associated sequence s, and assume that an infinitesimal change results in a different sequence, say s'. Then, there is an ordering $\mathcal{O}'$ of edges and facilities such that even with $z = \theta$, the run of the algorithm results in the sequence s'. For instance, listing s' followed by an arbitrary ordering of the edges and facilities that did not go tight in s', suffices.

For $z = 0$, the algorithm, run with ordering $\mathcal{O}$, opens all facilities, and for $z = nc_{\max}$ it opens only one facility, where n is the total number of vertices (cities and facilities) and $c_{\max}$ is the length of the longest edge. Consider the number of facilities opened as a function of z. Each discontinuity in this function must occur at a critical z. In Lemma 10 we will show that two critical z's are separated by at least $c = 2^{-(poly(n)+L)}$, where L is the number of bits needed to represent the longest edge.

We will conduct a binary search on the interval $[0, nc_{\max}]$ to find z_2 and z_1 for which the algorithm opens $k_2 > k$ and $k_1 < k$ facilities respectively, and furthermore, $z_1 - z_2 < c$. Since c can be written in polynomially many bits, this can be done in polynomial time. Furthermore, by Lemma 10, there can be only one critical z in the interval $[z_2, z_1]$; let this be θ. Let $(\boldsymbol{x}^l, \boldsymbol{y}^l)$ and $(\boldsymbol{x}^s, \boldsymbol{y}^s)$ be the primal solutions found by the algorithm at z_2 and z_1 respectively. Then, by the argument given above, there are orderings of edges and facilities under which the algorithm produces these solutions even for $z = \theta$. These two primal solutions, which can be found in polynomial time, have all the promised properties. Notice that we did not have to explicitly find θ.

Lemma 10 *Two critical z's are separated by at least $c = 2^{-(poly(n)+L)}$, where L is the number of bits needed to represent the longest edge.*

Proof : Let θ be a critical z. Assume that increasing z

slightly beyond θ results in a different sequence; the other case is analogous. Let this different sequence be s. Observe that $\theta = \inf_z\{z : z \text{ gives the sequence s}\}$. Consider values of z that give s as the sequence. We will show below that this set is the feasible region of a polynomial sized linear program. Therefore, θ can be written using $\log_2(1/c)$ bits.

Let $t_1, t_2, \ldots$ be variables representing times at which events happen in sequence s. For any time t_i, we know the events that have happened so far, and so the dual variables α_j and β_{ij} can be written in terms of $t_1, \ldots t_i$. The linear program will have three types of constraints for time t_i (involving the variables $t_1, \ldots t_i$ and z):

- The edge or facility represented by event s_i is tight.
- Any edge or facility coming earlier in ordering $\mathcal{O}$ than the edge or facility represented by event s_i is not tight.
- For all other edges and facilities, we have feasibility.

Additionally, we include the constraints $t_1 \le t_2 \le \cdots$. □

3.3. Randomized rounding

Let us show how to round $(\boldsymbol{x}, \boldsymbol{y})$ into an integral solution to the k-median problem, in the process at most doubling the cost.

If $a \ge 1/2$, then the solution $(\boldsymbol{x}^s, \boldsymbol{y}^s)$ satisfies these requirements, since

$$\sum_{i\in F,\, j\in C} c_{ij} x^s_{ij} \le (1/a)\left(\sum_{i\in F,\, j\in C} c_{ij} x_{ij}\right) \le 2\left(\sum_{i\in F,\, j\in C} c_{ij} x_{ij}\right).$$

So, for the remaining discussion, assume that $a < 1/2$; this implies that $a < b$. Denote by A and B the set of facilities opened in the solutions $\boldsymbol{y}^s$ and $\boldsymbol{y}^l$ respectively. We will open k facilities from B using the following algorithm: Order the facilities in A arbitrarily. For each facility $a \in A$, in this order, open the closest unopened facility of B; this step *deterministically* opens a total of k_1 facilities. Of the remaining $k_2 - k_1$ facilities of B, open a randomly chosen subset of size $k - k_1$; notice that the probability of a facility being opened is $(k-k_1)/(k_2-k_1) = b$. Let $I \subset B$ be the set of open facilities; $|I| = k$. Notice that if facility $i \in A \cap B$, then i will be opened by the algorithm.

The map ϕ from C to I is specified as follows: Let city j be connected to facilities i_1 and i_2 in $\boldsymbol{x}^s$ and $\boldsymbol{x}^l$ respectively. If $i_2 \in I$, then $\phi(j) = i_2$. Otherwise, $\phi(j) = i_3$, where i_3 is the facility in B opened due to $i_1 \in A$. Denote by $\text{cost}(j)$ the connection cost for city j in the fracitonal solution $(\boldsymbol{x}, \boldsymbol{y})$; $\text{cost}(j) = ac_{i_1j} + bc_{i_2j}$.

Lemma 11 *If $a < 1/2$, the expected connection cost for city j in the integral solution, $E(c_{\phi(j)j})$, is at most $2\text{cost}(j)$. Moreover, $E(c_{\phi(j)j})$ can be efficiently computed.*

Proof : Let us consider two cases. The first case is that i_2 was picked deterministically. If so, $\phi(j) = i_2$. Furthermore,

$$c_{i_2j} \le \frac{1}{b}\text{cost}(j) < 2cost(j),$$

since $b > 1/2$.

The second case is that i_2 was not picked deterministically. Now, with probability b, $\phi(j) = i_2$, and with probability a, $\phi(j) = i_3$, where i_3 was the facility deterministically opened by i_1. Since i_2 was not opened deterministically by i_1, $c_{i_1i_3} \le c_{i_1i_2}$. By the triangle inequality, $c_{i_1i_2} \le c_{i_1j} + c_{i_2j}$, and

$$c_{i_3j} \le c_{i_1i_3} + c_{i_1j} \le 2c_{i_1j} + c_{i_2j}.$$

Finally,

$$\begin{aligned} c_{\phi(j)j} \le ac_{i_3j} + bc_{i_2j} &\le a(2c_{i_1j} + c_{i_2j}) + bc_{i_2j} \\ &\le 2(ac_{i_1j} + bc_{i_2j}) = 2\text{cost}(j), \end{aligned}$$

where the last inequality follows from the fact that $b > a$.

Clearly, in both cases, $E(c_{\phi(j)j})$ is easy to compute. □

Let $(\boldsymbol{x}^k, \boldsymbol{y}^k)$ denote the integral solution obtained to the k-median problem by this randomized rounding procedure. Then,

Lemma 12

$$E(\sum_{i\in F,\, j\in C} c_{ij} x^k_{ij}) \le 2(\sum_{i\in F,\, j\in C} c_{ij} x_{ij}),$$

and moreover, the expected cost of the solution found can be computed efficiently.

3.4. Derandomization

The derandomization follows in a straightforward manner using the method of conditional expectation. Let $B' \subset B$, $|B'| = k_2 - k_1$, be the set of facilities of B that are not opened by the deterministic step. For a choice $D \subset B'$, $|D| \le k - k_1$, denote by $E(D, B' - D)$ the expected cost of the solution if all facilities in D and $(B - B')$ are opened and additionally $k - k_1 - |D|$ facilities are randomly opened from $B' - D$. Since each facility of $B' - D$ is equally likely to be opened, we get

$$E(D, B' - D) = \frac{1}{|B' - D|} \sum_{i\in B'-D} E(D \cup \{i\}, B' - (D \cup \{i\})).$$

This implies that there is an i such that $E(D \cup \{i\}, B' - (D \cup \{i\})) \leq E(B', D)$. Choose such an i and replace D by $D \cup \{i\}$. Notice that the computation of $E(D \cup \{i\}, B' - (D \cup \{i\}))$ can be done as in Lemma 12.

3.5. Improving the running time

The algorithm given above has a large running time because of the expensive binary search that needs to be carried out until the difference between z_1 and z_2 was inverse exponential, as required by Lemma 10. The next lemma shows that at the expense of a slight deterioration in the approximation guarantee, the binary search can be stopped when the difference between z_1 and z_2 is inverse polynomial.

Let us carry out the binary search until we find z_2 and z_1 for which the algorithm opens $k_2 > k$ and $k_1 < k$ facilities respectively, and $z_1 - z_2 \leq (c_{\min}/12{n_c}^2)$, where $c_{\min}$ is the length of the shortest edge. As before, let $(\boldsymbol{x}^s, \boldsymbol{y}^s)$ and $(\boldsymbol{x}^l, \boldsymbol{y}^l)$ be the primal solutions found at z_1 and z_2 respectively, and let $(\boldsymbol{\alpha}^s, \boldsymbol{\beta}^s)$ and $(\boldsymbol{\alpha}^l, \boldsymbol{\beta}^l)$ be the corresponding dual solutions found. Further, let $(\boldsymbol{x}, \boldsymbol{y})$ be the convex combination, with multipliers a and b, of the primal solutions that fractionally opens k facilities. This is a fractional feasible solution to the k-median problem in which each city is connected to at most two facilities.

Lemma 13 *The cost of $(\boldsymbol{x}, \boldsymbol{y})$ is within a factor of $(3+1/n_c)$ of the cost of an optimal fractional solution to the k-median problem.*

Proof : By Theorem 7 we have:

$$\sum_{i \in F,\ j \in C} c_{ij} x_{ij}^s \ \leq\ 3(\sum_{j \in C} \alpha_j^s - z_1 k_1),$$

and

$$\sum_{i \in F,\ j \in C} c_{ij} x_{ij}^l \ \leq\ 3(\sum_{j \in C} \alpha_j^l - z_2 k_2).$$

Since $z_1 > z_2$, $(\boldsymbol{\alpha}^l, \boldsymbol{\beta}^l)$ is a feasible dual solution to the facility location problem even if the cost of facilities is z_1. We would like to replace z_2 by z_1 in the second inequality, at the expense of the increased factor. This is achieved using the upper bound on $z_1 - z_2$, and the fact that $\sum_{i \in F,\ j \in C} c_{ij} x_{ij}^l \ \geq\ c_{\min}$. We get:

$$\sum_{i \in F,\ j \in C} c_{ij} x_{ij}^l \ \leq\ (3 + \frac{1}{n_c})(\sum_{j \in C} \alpha_j^l - z_1 k_2).$$

Multiplying this inequality by b and the first inequality by a and adding, we get

$$\sum_{i \in F,\ j \in C} c_{ij} x_{ij} \ \leq\ (3 + \frac{1}{n_c})(\sum_{j \in C} \alpha_j - z_1 k),$$

where $\boldsymbol{\alpha} = a\boldsymbol{\alpha}^s + b\boldsymbol{\alpha}^l$. Let $\boldsymbol{\beta} = a\boldsymbol{\beta}^s + b\boldsymbol{\beta}^l$. Observe that $(\boldsymbol{\alpha}, \boldsymbol{\beta}, z_1)$ is a feasible solution to the dual of the k-median problem. The lemma follows. □

Next, we give an improved randomized rounding procedure that produces an integral solution to the k-median problem from $(\boldsymbol{x}, \boldsymbol{y})$, in the process incurring a multiplicative factor of $1 + \max(a, b)$. It is easy to see that $a \leq 1 - 1/n_c$ (this happens for $k_1 = k - 1$ and $k_2 = n_c$) and $b \leq 1 - 1/k$ (this happens for $k_1 = 1$ and $k_2 = k + 1$). Therefore, $1 + \max(a, b) \leq 2 - 1/n_c$.

Let A and B be the sets of facilities opened in the two solutions, $|A| = k_1$ and $|B| = k_2$. First, open facilities in $A \cap B$. Order the remaining facilities of A arbitrarily, and for each facility $i \in A - B$ in this order, pair it with the closest unpaired facility of $B - A$. Order these k_1 pairs according to the order of their first elements. Let B' denote the unpaired facilities of $B - A$, $|B'| = k_2 - k_1$. >From each pair $(i_1, i_2), i_1 \in (A - B), i_2 \in (B - A)$, one facility is opened: i_1 with probability a and i_2 with probability b. In addition, a set of cardinality $k - k_1$ is picked randomly from B' and facilities in this set are opened. Notice that each facility in B' has a probability of b of being opened. Let I be the set of facilities opened, $|I| = k$.

The function $\phi : C \to I$ is defined as follows: Consider city j, and suppose that it is connected to $i_1 \in A$ and $i_2 \in B$ in the two solutions. If (i_1, i_2) is a pair, j is connected to the facility that is opened from this pair. Otherwise, there are two cases. First, that i_2 lies in an earlier numbered pair than i_1, say the pair $(i_3, i_2), i_3 \in A$. In this case, if $i_1 \in I$, then j is connected to i_1; otherwise, it is connected to i_2 or i_3, whichever is opened. In the second case, let the pair containing i_1 be $(i_1, i_3), i_3 \in B$. Now, if $i_2 \in I$, then j is connected to i_2; otherwise, it is connected to i_1 or i_3, whichever is opened.

One can prove, using the ideas of Lemma 11, that the expected cost for connecting city j goes up by a factor of at most $1 + \max(a, b)$. Let us show this for the case that $i_2 \in B'$, which falls in the last case given in the algorithm. As before, $\text{cost}(j) = a c_{i_1 j} + b c_{i_2 j}$. By the procedure given above,

$$E(c_{\phi(j)j}) = b c_{i_2 j} + a(a c_{i_1 j} + b c_{i_3 j}).$$

This follows from the fact that j is connected to i_2 with probability b, the probability that i_2 is opened; i_2 is not opened with probability a, and in this case j is connected to i_1 with probability a and to i_3 with probability b. Since i_2 is left unpaired, $c_{i_1 i_3} \leq c_{i_1 i_2}$. By the triangle inequality,

$$c_{i_3 j} \leq c_{i_1 j} + c_{i_1 i_3} \leq 2c_{i_1 j} + c_{i_1 i_2}.$$

Substituting, we get

$$E(c_{\phi(j)j}) \ \ \leq b(1 + a)c_{i_2 j} + a(1 + b)c_{i_1 j}$$

$$\leq (1 + \max(a, b))(ac_{i_1 j} + bc_{i_2 j})$$
$$= (1 + \max(a, b))\text{cost}(j).$$

Altogether, the approximation guarantee we get for the k-median problem is $(2 - 1/n_c)(3 + 1/n_c) < 6$. Using the method of conditional probabilities, this procedure can be derandomized, as in Section 3.4. The binary search will make $O(\log_2(n^3 c_{\max}/c_{\min})) = O(L + \log n)$ probes. The running time for each probe is dominated by the time taken to run Algorithm 1; randomized rounding takes $O(n)$ time and derandomization takes $O(m)$ time. Hence we get:

Theorem 14 *The algorithm given above achieves an approximation factor of 6 for the k-median problem, and has a running time of $O(m \log m(L + \log(n)))$.*

3.6. Tight example

We do not have a tight example of factor 6 for the complete k-median algorithm. However, we give below an infinite family of instances which show that the analysis of the two randomized rounding procedures given in Sections 3.3 and 3.5 cannot be improved.

The two solutions $(\boldsymbol{x}^s, \boldsymbol{y}^s)$ and $(\boldsymbol{x}^l, \boldsymbol{y}^l)$ open one facility, f_0, and $k+1$ facilities, $f_1, \ldots, f_{k+1}$ respectively. The distance between f_0 and any other f_i is 1, and that between two facilities in the second set is 2. All n cities are at a distance of ϵ from f_{k+1}. The rest of the distances are given by triangle inequality. The convex combination is constructed with $a = 1/k$ and $b = 1 - 1/k$.

Now, the cost of the convex combination is $an + b\epsilon n$. The expected cost of the solutions produced by the two randomized rounding procedures is $2an + b\epsilon n$ and $a(1+b)n + b\epsilon n$. So, letting ϵ tend to 0 gives the tight examples.

4. A common generalization of the two problems

Consider the uncapacitated facility location problem with the additional constraint that at most k facilities can be opened. This is a common generalization of the two problems solved in this paper – if k is made n_f, we get the first problem and if the facility costs are set to zero, we get the second problem.

The techniques of this paper yield a factor 6 algorithm for this generalization as well. The high level idea is as follows: We will first remove the restriction that at most k facilities be opened, and instead set the cost of opening each facility i to $f_i + z$. Now, binary search on z will yield two values of z, close to each other, for which Algorithm 1 opens $k_1 < k$ and $k_2 > k$ facilities respectively. An appropriate convex combination of these two solutions gives a fractional solution that opens exactly k facilities, with the additional property that each city is connected to at most two facilities. The cost of this solution is within thrice the cost of an optimal fractional solution. Notice that the randomized rounding procedure given in Section 3.3 is not applicable to this setting since it opens facilities only in set B, and these could have high costs. However, the procedure given in Section 3.5 gets around this issue – it ensures that the expected cost of opening facilities in the rounded solution is the same as the cost of opening facilities in the convex combination. Finally, the derandomization procedure can also be carried out in this setting.

Theorem 15 *There is a factor 6 approximation algorithm for common generalization of uncapacitated facility location and k-median problems in which facilities have costs and at most k of them can be opened.*

5. Dealing with capacities

We consider the following variant of the capacitated metric facility location problem: each facility can be opened an unbounded number of times; if facility i is opened y_i times, it can serve at most $u_i y_i$ cities. The LP-relaxation of this problem has the following extra constraint:

$$\forall i \in F: \; u_i y_i - \sum_{j \in C} x_{ij} \geq 0.$$

Let the dual variable corresponding to this constraint be γ_i. Then, the dual program is:

$$\begin{aligned} \text{maximize} \quad & \sum_{j \in C} \alpha_j & (7) \\ \text{subject to} \quad & \forall i \in F, j \in C: \; \alpha_j - \beta_{ij} - \gamma_i \leq c_{ij} \\ & \forall i \in F: \; u_i \gamma_i + \sum_{j \in C} \beta_{ij} \leq f_i \\ & \forall j \in C: \; \alpha_j \geq 0 \\ & \forall i \in F: \; \gamma_i \geq 0 \\ & \forall i \in F, j \in C: \; \beta_{ij} \geq 0 \end{aligned}$$

For each facility i, let us fix $\gamma_i = \frac{3f_i}{4u_i}$. This step enables us to get rid of the variables γ_i from LP (7), and the resulting linear program is again the dual of an uncapacitated facility location problem. The primal program for this modified dual is:

$$\begin{aligned} \text{minimize} \quad & \sum_{i \in F, j \in C} (c_{ij} + \frac{3f_i}{4u_i}) x_{ij} + \sum_{i \in F} \frac{f_i}{4} Y_i & (8) \\ \text{subject to} \quad & \forall j \in C: \; \sum_{i \in F} x_{ij} \geq 1 \\ & \forall i \in F, j \in C: \; Y_i - x_{ij} \geq 0 \end{aligned}$$

$$\forall i \in F, j \in C: \; x_{ij} \geq 0$$
$$\forall i \in F: \; Y_i \geq 0$$

It is easy to see that $c_{ij} + \frac{3f_i}{4u_i}$ still satisfies the triangle inequality. Using Algorithm 1, we can now find a 0/1 integral solution to this LP satisfying

$$\sum_{i \in F, j \in C} (c_{ij} + \frac{3f_i}{4u_i}) x_{ij} + 3 \sum_{i \in F} \frac{f_i}{4} Y_i \leq 3 \sum_{j \in C} \alpha_j,$$

by Theorem 7. Now, our solution to the capacitated problem is: x_{ij}'s are as in this solution, and $y_i = \lceil \frac{\sum_{j \in C} x_{ij}}{u_i} \rceil$. This gives the following relationship between y_i and Y_i:

$$y_i \leq Y_i + \frac{\sum_{j \in C} x_{ij}}{u_i}.$$

Using this relationship and the above inequality we get:

$$\sum_{i \in F, j \in C} c_{ij} x_{ij} + \frac{3}{4} \sum_{i \in F} f_i y_i \leq 3 \sum_{j \in C} \alpha_j.$$

This implies

$$\sum_{i \in F, j \in C} c_{ij} x_{ij} + \sum_{i \in F} f_i y_i \leq 4 \sum_{j \in C} \alpha_j,$$

thereby giving an approximation guarantee of factor 4.

6. Discussion

A large fraction of the theory of approximation algorithms, as we know it today, is built around linear programming, which provides two main algorithm design techniques: rounding and the primal-dual schema. Both techniques have yielded algorithms with good approximation guarantees, often achieving the integrality gap of the relaxation being used. However, with respect to the running times of the algorithms derived, the two methods differ widely. Rounding resorts to the "big hammer" approach of solving the linear program and therefore leads to inefficient algorithms. On the other hand, the primal-dual schema leaves enough room to exploit the special combinatorial structure of individual problems and has therefore lead to efficient algorithms. Once the algorithm is obtained, typically the scaffolding of linear programming can be completely dispensed with to obtain a purely combinatorial algorithm. As was done in this paper, it seems worthwhile examining various algorithms derived using rounding, to see if efficient combinatorial algorithms achieving the same factors can be obtained.

It is instructive to compare the current status of primal-dual approximation algorithms with the (mature) status of exact primal-dual algorithms. In the latter setting, only one underlying mechanism is used: iteratively ensuring all complementary slackness conditions. On termination, an optimal (integral) solution to the LP is obtained. In the former setting, we are not seeking an optimal solution to the LP (since the LP may not have any optimal integral solutions), and so there is a need to introduce a further relaxation. Relaxing complementary slackness conditions (which itself can be carried out in more than one way) is only one of the possibilities (see [30] for an alternative mechanism). Another point of difference is that in the exact setting, more sophisticated dual growth algorithms have been given, e.g. [15]. In the approximation setting, other than [30], all primal-dual algorithms use a simple greedy dual growth algorithm. It appears that the full potential of this potent technique has yet to be exploited in the approximation setting.

At a more detailed level, the issue of modifying Algorithm 1 so it opens exactly k facilities deserves some thought – this is a possible avenue for improving the factor for the k-median problem. It would be nice to improve the running time of the facility location algorithm in case the metric is specified as the closure of a sparse graph, rather than a complete bipartite graph. Another question is to obtain a non-trivial approximation algorithm for the capacitated facility location problem.

7. Acknowledgement

We wish to thank Pete Veinott for interesting discussions.

References

[1] A. Agrawal, P. Klein, and R. Ravi. When trees collide: An approximation algorithm for the generalized Steiner problem on networks. *SIAM J. on Computing*, 24:440-456, 1995.

[2] S. Arora, P. Raghavan, and S. Rao. Approximation schemes for Euclidean k-medians and related problems.*Proc. 30th ACM Symp. on Theory of Computing*, 106-113, 1998.

[3] M. L. Balinksi. On finding integer solutions to linear programs. *Proc. IBM Scientific Computing Symp. on Combinatorial Problems*, 225-248. IBM, 1966.

[4] J. Bar-Ilan, G. Kortsarz, and D. Peleg. How to allocate network centers. *J. Algorithms*, 15:385-415, 1993.

[5] Y. Bartal. Probabilistic approximation of metric spaces and its algorithmic applications. *Proc. 37th IEEE Symp. on Foundation of Computer Science*, 184-193, 1996.

[6] R. Bar-Yehuda and S. Even. A linear time approximation algorihtm for the weighted vertex cover problem. *Journal of Algorithms*, 2:198-203, 1981.

[7] P. S. Bradley, U. M. Fayyad, and O. L. Mangasarian. Mathematical programming for data mining: formulations and Challenges. Microsoft Technical Report, January 1998.

[8] M. Charikar, C. Chekuri, A. Goel, and S. Guha. Rounding via trees: deterministic approximation algorithms for group Steiner trees and k-median. *Proc. 30th ACM Symp. on Theory of Computing*, 114-123, 1998.

[9] M. Charikar, S. Guha, E. Tardos, and D. B. Shmoys. A constant-factor approximation algorithm for the k-median problem. To appear in *Proc. 31st ACM Symp. on Theory of Computing*, 1999.

[10] G. Cornuejols, G. L. Nemhauser, and L. A. Wolsey. The uncapacitated facility location problem. In P. Mirchandani and R. Francis, eds., *Discrete Location Theory*. John Wiley and Sons, New York, 1990.

[11] F. Chudak. Improved approximation algorithms for uncapacitated facility location. In R.E. Bixby, E.A. Boyd and R.Z. Rios-Mercado, eds., *Integer Programming and Combinatorial Optimization*, Springer LNCS Vol. 1412, 180-194, 1998.

[12] F. Chudak and D. Shmoys. Improved approximation algorithms for the uncapacitated facility location problem. Unpublished manuscript, 1998.

[13] F. Chudak and D. Shmoys. Improved approximation algorithms for the capacitated facility location problem. *Proc. 10th ACM-SIAM Symp. on Discrete Algorithms*, S875-S876, 1999.

[14] F. A. Chudap and D. P. Williamson. Improved approximation algorithms for capacitated facility location problems. To appear in IPCO 1999.

[15] J. Edmonds. Maximum matching and a polyhedron with 0,1-vertices. *Journal of Research of the National Bureau of Standards B*, 69B:125-130, 1965.

[16] N. Garg, V. Vazirani, and M. Yannakakis. Primal-dual approximation algorithms for integral flow in multicut in trees, with application to matching and set cover. *Proc. 20th International Colloquium on Automata, Languages and Programming*, 1993.

[17] M. Goemans, A. Goldberg, S. Plotkin, D. Shmoys, E. Tardos, and D. Williamson. Improved approximation algorithms for network design problems. *Proc. 5th ACM-SIAM Symp. on Discrete Algorithms*, 223-232, 1994.

[18] M. X. Goemans, D. P. Williamson. A general approximation technique for constrained forest problems. *SIAM Journal of Computing*, 24:296-317, 1995.

[19] S. Guha and S. Khuller. Greedy strikes back: Improved facility location algorithms. *Proc. 9th ACM-SIAM Symp. on Discrete Algorithms*, 649-657, 1998.

[20] D. S. Hochbaum. Heuristics for the fixed cost median problem. *Math. Programming*, 22:148-162, 1982.

[21] K. Jain, Ion Măndoiu, Vijay V. Vazirani, and David P. Willamson. A primal-dual schema based approximation algorithm for the element connectivity problem. *Proc. 10th Symp. on Discrete Algorithms*, 484-489, 1999.

[22] L. Kaufman, M. vanden Eede, and P. Hansen. A plant and warehouse location problem. *Operational Research Quarterly*, 28:547-557, 1977.

[23] S. Khuller and Y. J. Sussmann. The capacitated k-center problem. *Proc. 4th European Symp. on Algorithms, Lecture Notes in Computer Science 1136*, 152-166, Berlin, 1996. Springer.

[24] M. R. Korupolu, C. G. Plaxton, and R. Rajaraman. Analysis of a local search heuristic for facility location problems. *Proc. 9th ACM-SIAM Symp. on Discrete Algorithms*, 1-10, 1998.

[25] A. A. Kuehn and M. J. Hamburger. A heuristic program for locating warehouses. *Management Sci.,* 9:643-666, 1963.

[26] J.-H. Lin and J. S. Vitter. Approximation algorithms for geometric median problems. *Inform. Proc. Lett.,* 44:245-249, 1992.

[27] J.-H. Lin and J. S. Vitter. ϵ-approximation with minimum packing constraint violation. *Proc. 24th ACM Symp. on Theory of Computing*,771-782, 1992.

[28] P. B. Mirchandani and R. L. Francis, eds. *Discrete Location Theory.* John Wiley and Sons, New York, 1990.

[29] G. L. Nemhauser and L. A. Wolsey. *Integer and Combinatorial Optimization*. John Wiley and Sons, New York, 1990.

[30] S. Rajagopalan and V. V. Vazirani, On the Bidirected Cut Relaxation for the Metric Steiner Tree Problem. *Proc. 10th ACM-SIAM Symp. on Discrete Algorithms*, 742-751, (1999).

[31] R. Ravi and D. Williamson. An approximation algorithm for minimum-cost vertex-connectivity problems. *Proc. 6th ACM Symp. on Theory of Computing,* 398-419, 1980.

[32] D. B. Shmoys, E. Tardos, and K. I. Aardal. Approximation algorithms for facility location problems. *Proc. 29th ACM Symp. on Theory of Computing,* 265-274, 1997.

[33] J. F. Stollsteimer. *The effect of technical change and output expansion on the optimum number, size and location of pear marketing facilities in a California pear producing region.* PhD thesis, University of California at Berkeley, Berkeley, California, 1961.

[34] J. F. Stollsteimer. A working model for plant numbers and locations. *J. Farm Econom.*, 45:631-645, 1963.

[35] M. Sviridenko. Personal communication, July, 1998.

[36] D. P. Williamson, M. X. Goemans, M. Mihail, and V. V. Vazirani. A primal-dual approximation algorithm for generalized Steiner network problems. *Combinatorica*, 15:435-454, December 1995.