Proc. 12th ACM-SIAM Sympos. Discrete Algorithms, pp. 642-651, Washington, DC, 2001.

Algorithms for Facility Location Problems with Outliers

(Extended Abstract)

Moses Charikar[*] Samir Khuller[†] David M. Mount[‡] Giri Narasimhan[§]

Abstract

Facility location problems are traditionally investigated with the assumption that *all* the clients are to be provided service. A significant shortcoming of this formulation is that a few very distant clients, called *outliers*, can exert a disproportionately strong influence over the final solution. In this paper we explore a generalization of various facility location problems (K-center, K-median, uncapacitated facility location etc) to the case when only a specified fraction of the customers are to be served. What makes the problems harder is that we have to also select the subset that should get service. We provide generalizations of various approximation algorithms to deal with this added constraint.

1 Introduction

The facility location problem and the related clustering problems, k-median and k-center, are widely studied in operations research and computer science [3, 7, 22, 24, 32]. Typically in problems of this type, we are given an n-vertex graph whose edge weights define a distance metric. Let c_{ij} denote the distance between nodes i and j.

In the (uncapacitated) facility location problem, each node i is associated with a *facility cost* f_i, which reflects the cost of opening a facility at this node. The problem is to open a subset of facilities so as to minimize the sum of facility costs and the *service cost*, which is defined to be the sum of distances from each node to its closest open facility. The k-median problem differs in that exactly k facilities be opened, and there is no facility cost, only service cost. The k-center problem differs from the k-median problem in that the service cost is defined to be the maximum distance (rather than the sum of distances) from any facility to its closest open facility. All of these optimization problems are NP-hard, and polynomial time approximation algorithms have been studied (see [17, 4, 5, 6, 31, 19, 13, 15, 21, 23]).

A significant shortcoming of these simple formulations is that a few very distant clients, called *outliers*, can exert a disproportionately strong influence over the final solution. This is clearly true for min-max problems like k-centers, where a single client residing far from the other clients may force a center to be placed in its vicinity. With min-sum formulations this effect is reduced, but it is still possible if the outliers are sufficiently far away. Such *outliers* have the undesirable effect of increasing the cost of the solution, without improving the level of service to the majority of clients.

For many applications of facility location, such as mail delivery, it may be that all clients must be serviced. However, for the majority of commercial applications of facility location, it may be economically essential to ignore very distant outliers. For example, Kmart claims to provide service to 88% of the US population within a radius of 6 miles using the current set of Kmart stores [14]. Clearly, if they tried to cover the entire population of the country, either the covering radius or the number of stores would need to be significantly higher. The simple formulations of facility location problems described above can lead to nonsensical solutions, in which facilities are placed in isolated areas just to satisfy the demands of a few outliers. Remarkably, we know if virtually no existing work on this important variant of the problem.

In this paper we introduce the notion of how to perform facility location in contexts where outliers may exist. Our principal contribution is to formulate variations of the facility location problems so that outliers can be handled in a meaningful way, and to consider the computational consequences of these new

[*] Department of Computer Science, Stanford University, Stanford, CA 94305. Research supported by the Pierre and Christine Lamond Fellowship, and ARO MURI Grant DAAH04-96-1-0007 and NSF Grant IIS-9811904. E-mail : moses@cs.stanford.edu

[†] Department of Computer Science and Institute for Advanced Computer Studies, University of Maryland, College Park, MD 20742. Research supported by NSF CAREER Award CCR-9501355 and NSF Award CCR-9820965. E-mail : samir@cs.umd.edu.

[‡] Department of Computer Science and Institute for Advanced Computer Studies, University of Maryland, College Park, MD 20742. Research supported by NSF Award CCR-9712379. E-mail : mount@cs.umd.edu.

[§] Mathematical Sciences Department, University of Memphis, Memphis TN 38152. This work was done while this author was visiting UMIACS. E-mail : giri@msci.memphis.edu.

formulations. The essential feature of our formulations is to provide additional parameters that allow a small subset of the clients to be denied service, thereby reducing costs drastically. These denied clients do not contribute to the final service cost. We propose two ways to do this.

Robust facility location: In addition to the standard problem formulation we are given a parameter p. The problem is to place facilities so as to minimize the service cost to any subset of facilities of size at least p. (Recall that there are n clients.) Setting $p = n$ is equivalent to the standard formulation.

Facility location with penalties: Each client j is associated with a penalty p_j. For each client we may decide to either provide service, and pay the service cost to its nearest facility or to pay the penalty. Setting the penalties to ∞ gives the standard formulation. (This notion has been studied earlier in the context of TSP and Steiner trees, see [11, 10] and references therein.)

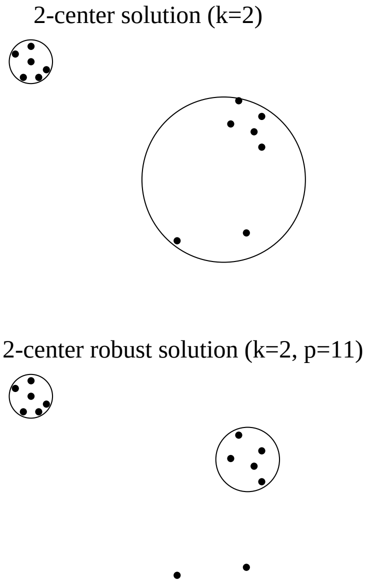

Figure 1: Example to illustrate how robust measures lead to better clustering solutions.

These two modifications can be applied to any of the variants of the facility location problems. The terminology used in the first formulation is borrowed from the well-studied area of robust estimators in statistics [30]. An estimator is said to be *robust* with a *breakdown point* of α if the estimator's value is immune to the placement of any fraction of up to α outliers, no matter how far they lie from the rest of the population. In our case $p = \lceil (1-\alpha)n \rceil$. For facility location, α represents the fraction of the population to which we are willing to deny service. An example is shown in Figure 1.

Note that by using the notion of "weights" in the K-center problem one can reduce the effect of outliers [12, 29] and design constant factor approximation algorithms. (The definition of the distance function is modified to be a weighted distance function, by multiplying the weight of the node with its distance to the closest center.) The main problem with this approach is that we need to "know" who the outliers are before running the algorithm and we can decrease their effect on the objective function. Our approach on the other hand, identifies the outliers automatically.

The *robust k-center problem with forbidden centers* is the same as the above problem with the added constraint that some points are forbidden from being chosen as centers. The *robust k-supplier* problem requires that k centers be selected from a given set of suppliers so that the maximum distance from p clients to the chosen centers is minimized. We report the following results for the three min-max clustering problems defined above:

1.1 Robust K-Center Results.

The robust k-center problem is of interest in clustering applications in data analysis, where erroneous data is present or where it is desirable to ignore less significant subsets of the data [18].

Lower Bounds: If the input points are in $\Re^d$ (with the Euclidean metric), then it is NP-hard to approximate the optimal *cost* (for all variants) to within a factor of 1.822. If the input points in V are from an arbitrary metric space, then unless P = NP, the optimal *cost* for the robust k-center problem with forbidden centers cannot be approximated to within a factor of $3 - \epsilon$.

Arbitrary Metric: There exist $O(n^3)$-time 3-approximate algorithms for all three robust clustering problems.

1.2 Facility Location and K-Median problem with penalties.

We introduce variants of the k-median and facility location problems. The variant involves associating each vertex j with a penalty p_j. In the original k-median problem, we must choose k vertices as centers. Every vertex is then assigned to the closest center and the objective function is the sum of distances of vertices from their centers. In the k-median problem with penalties, we have the option of either connecting a vertex to a center or selecting it to be an outlier and not connecting it to any center. If we connect a particular vertex j, the contribution to the objective function is the distance of the vertex to the associated center. If vertex j is selected to be an outlier, the contribution to the objective function is the penalty p_j.

Similarly, in the facility location problem with penalties, each vertex can either be connected to a facility or selected to be an outlier. The objective function is the sum of the facility costs for all the open facilities, the assignment cost for the vertices that are connected to open facilities and the penalties for the vertices that are selected to be outliers.

Facility location with penalties: We can obtain a 3 approximation for the facility location problem with penalties in polynomial time.

k-median with penalties: We can obtain a 4 approximation for the k-median problem with penalties in polynomial time.

1.3 Robust Facility Location and K-Median Results. The above two variants of facility location and k-median with penalties are important building blocks in obtaining algorithms for the corresponding clustering problems with an upper bound on the number of excluded outliers.

Robust facility location: For the robust facility location problem we can obtain a 3 approximation in polynomial time.

Robust k-Median problem: We obtain a solution which has the number of outliers to be at most $(1+\epsilon)$ times the number of specified outliers and has cost at most $4(1+1/\epsilon)$ times the cost of the optimal solution.

1.4 Extensions. In addition, we can obtain a 2-approximation for the robust bottleneck-TSP problem as well. Rather than asking for a min-max tour visiting all n locations, we would like to visit only a specified number of locations.

For Euclidean instances of problems such as k-TSP [1], k-medians [2], etc., we note that the PTAS's can all be extended to obtain PTAS's for robust versions of these problems.

2 Lower Bounds

Feder and Greene [8] showed that if k is part of the input, then the k-center problem (i.e., the non-robust version) is NP-hard to approximate to within a factor of 1.822 even for points in the plane (with the Euclidean distance metric). We observe that the proof can be easily extended to a similar claim about the robust k-center problem for any given value of p.

Since the robust k-center problem is a generalization of the (non-robust) k-center, the lower bound of $2-\epsilon$ (assuming $\mathrm{P} \neq \mathrm{NP}$) for the approximation ratio mentioned in [16] also holds for the robust case. The following theorem proves a hardness of $3-\epsilon$ on the performance of any approximation algorithm for the robust k-center problem (with forbidden centers) for input points from an arbitrary metric space.

Note that the non-robust version of the k-center problem with forbidden centers is a special case of the *cost* k-center problem [17] for which such a lower bound is not known.

Theorem 2.1. *For input points from an arbitrary metric space, the optimal cost for the robust k-center problem with forbidden centers cannot be approximated to within a factor of $3-\epsilon$ unless $\mathrm{P} = \mathrm{NP}$.*

Proof. Our proof uses a result on the *maximum coverage problem* proved independently by Feige [9] and Khuller *et al.* [20]. The problem is defined as follows: given a collection S of sets over a domain B of elements, finding a subset $S' \subseteq S$ of cardinality k that maximizes the number of elements covered from B is a NP-hard problem. It was shown that it cannot be approximated to a factor of $\alpha = 1 - 1/e + \epsilon$ for any $\epsilon > 0$ and that this result is the best possible unless $\mathrm{P} = \mathrm{NP}$ [9].

Consider an instance of the maximum coverage problem. Let S be a collection of m sets whose domain is a set B with n elements. We assume that we are asked to find k sets that cover the maximum number of elements from B.

We construct an instance of the robust k-center problem with forbidden centers as follows: We construct a set B' of size $nq+1$ from B by making q "copies" of each of the n elements and then adding one "special" element to the set. The value of q will be specified later. For each set $S_i \in S$, $1 \leq i \leq m$, we construct a set S'_i by including in it the special element and each copy of every element from S_i. Thus $|S'_i| = q|S_i| + 1$. Now construct a bipartite graph G with S' and B' as the two sets of vertices. There is an edge between $S'_i \in S'$ and $b \in B'$ iff $b \in S'_i$. Finally we set the vertices of B' as the forbidden centers, i.e., all centers must be chosen from S'. We assume that all edges of the graph have unit cost.

If we assume that there exist k sets in S that cover all n elements of B, then there exist k centers in S' that cover all $nq+1$ vertices of B' with radius 1, i.e., there exist k centers in S' that cover $nq+1+k$ vertices from G. If $p = nq+1+k$, then we know that a solution to the robust k-center problem with forbidden centers exists with *cost* equal to 1.

By the inapproximability result of the maximum coverage problem, we know that no polynomial-time algorithm can guarantee selecting k sets from S that cover more than αn elements from B (for convenience, we will avoid repeating the phrase "unless P = NP" until the end of the proof).

Suppose that there existed a polynomial-time algorithm that can guarantee selecting k centers from S' that cover p nodes with an approximation guarantee of $3 - \epsilon$. Note that any single center from S' can cover all vertices from S' with radius 2 because of the special element; however, the same center from S' will not be able to cover any more vertices from B' with radius $3 - \epsilon$ than it did with radius 1. Let m and x denote the numbers of nodes from S' and B', respectively, that are covered by the algorithm. Then $m + x \geq p = nq + 1 + k$. Thus $x \geq nq + 1 + k - m$. To get $x \geq \alpha(nq + 1)$, choose q such that $nq + 1 + k - m \geq \alpha(nq + 1)$. This implies that we need q such that the following holds

$$nq + 1 \geq \frac{m - k}{(1 - \alpha)}. \tag{2.1}$$

It is easy to see that q can be chosen so that this holds.

This implies that there is a polynomial time algorithm that can select k sets from S' that cover at least αn elements from B. This implies that $P = NP$.

Hence the proof. ∎

3 Robust Clustering for Arbitrary Metrics

We present a 3-approximation algorithm for the robust k-center problem with input points from an arbitrary metric. As with many approximation algorithms, the algorithm is simple, yet counter-intuitive, and has an interesting analysis of the performance ratio. We first present an algorithm that takes as input a radius r and a set of n points V from an arbitrary metric space and finds a solution S with $cost(S) \leq 3r$, assuming one exists with radius r.

Since the optimal *cost* must correspond to an interpoint distance, a 3-approximation to the optimal *cost* can then be found by simulating a binary search on the list of all approximate interpoint distances.

For each point $v_i \in V$, let G_i (E_i, resp.) denote the set of points that are within distance r ($3r$, resp.) from v_i. We refer to the sets G_i as *disks* of radius r and the sets E_i as the corresponding *expanded disks* of radius $3r$. We use the term *weight* of a disk (or expanded disk) to refer to its cardinality. The algorithm is as follows:

- Construct all disks and corresponding expanded disks.
- Repeat the following k times:
  - Let G_j be the heaviest disk, i.e. contains the most uncovered points.
  - Mark as covered all points in the corresponding expanded disk E_j.
  - Update all the disks and expanded disks, i.e., remove from them all covered points.
- If at least p points of V are marked as covered, then answer YES, else answer NO.

THEOREM 3.1. *Given a set V of n points from an arbitrary metric, an integer $k \leq n$, and an integer p, there exists a polynomial time 3-approximation algorithm for the robust k-center problem, the robust k-center problem with forbidden centers, and the robust k-suppliers problem.*

Proof. Let V be the n input points. Assume that $G_1, G_2, \ldots, G_k$ are the k disks selected in each of the k iterations. Let the disk centers be the points $v_1, v_2, \ldots, v_k$ and let the corresponding expanded disks be $E_1, E_2, \ldots, E_k$. Note that G_i (E_i, resp.) consists of a set of points that are within distance r ($3r$, resp.) from v_i. Let an optimal solution have centers at $v'_1, v'_2, \ldots, v'_k$ with the corresponding disks being $O_1, O_2, \ldots, O_k$. Consider an optimal solution consisting of the disks $O_1, O_2, \ldots, O_k$. The proof follows from showing that it is possible to order the optimum disks such that for each i, the first i expanded greedy disks $E_1 \cup E_2 \cup \ldots \cup E_i$ cover at least as many points as the first i optimal disks, $O_1 \cup O_2 \cup \ldots \cup O_i$. The proof is by induction using a charging argument that charges each point of the union of the optimal disks to a distinct point in the union of the expanded disks.

Assume that the disks $O_1, O_2, \ldots, O_{i-1}$ have been selected and that each point in their union has already been charged to a distinct point in $E_1 \cup E_2 \cup \ldots \cup E_{i-1}$, thus satisfying the induction hypothesis. Consider G_i. If $G_1 \cup G_2 \cup \ldots \cup G_i$ intersects any of the remaining $k - i + 1$ optimal disks, then let O_i be such a disk. Thus, $E_1 \cup E_2 \cup \ldots \cup E_i$ covers all the points of O_i. Charge each of the newly covered points of O_i to itself. Call this *charging rule* I. Each point can be charged only once (namely to itself) by this rule.

If, on the other hand, $G_1 \cup G_2 \cup \ldots \cup G_i$ does not intersect any of the remaining $k - i + 1$ optimal disks, then let *Unc* be the points that are still uncovered by by the first $i - 1$ expanded greedy disks, i.e.,

$$Unc = P - (E_1 \cup E_2 \cup \ldots \cup E_{i-1}).$$

Let O_i be the remaining optimal disk that covers the greatest number of points in *Unc*. Charge the points of O_i that have already been covered by expanded greedy disks to themselves (using charging rule I). Observe that by greediness, G_i covers at least as many elements of

Unc as O_i does. Charge these uncovered points of O_i to the uncovered points of G_i. Call this *charging rule* II. No future optimal disk will attempt to charge these same points by charging rule I, because G_i is disjoint of the remaining optimal disks. Also, no future optimal disk will attempt to charge points of G_i by charging rule II, since these will be charged to uncovered points of later greedy disks. Since $G_i \subseteq E_i$, we are done. ∎

Remark: We note that if the input points lie in $\Re^d$ with the Euclidean distance metric, then the above algorithm and proof can be modified to deal with a version of the k-center problem where the centers may or may not be located at the input points; the facility may also be placed at any other point in space. The main modification that is required is in determining the values of r to be searched. Searching through the list of interpoint distances is not sufficient any more. Instead we need to search all values of r such that the corresponding set of disks have at least three points on their boundary or have two points located diametrically opposite from each other. The time complexity in this case does increase, although it still remains polynomial.
Remark: If the points lie on a line, then the robust k center problem can be solved optimally using dynamic programming.

4 Linear programming relaxations

We now introduce LP relaxations and their duals for the penalty and robust versions of the facility location and k-median problems.

We use the following LP relaxation (LP1) for the facility location problem with penalties. We use $\mathbf{y}_i$ as a binary variable that is 1 if and only if facility i is opened at cost f_i. We use $\mathbf{x}_{ij}$ to denote that client j is assigned to facility i. The connecting cost is c_{ij} to connect client j to facility i. The penalty for not connecting a client is p_j. We use $\mathbf{r}_j$ as an indicator variable for whether or not client j is connected to a facility.

$$\min \sum_i f_i \cdot \mathbf{y}_i + \sum_{ij} c_{ij} \cdot \mathbf{x}_{ij} + \sum_j p_j \cdot \mathbf{r}_j \tag{4.2}$$

$$\forall\, ij \quad \mathbf{x}_{ij} \le \mathbf{y}_i \tag{4.3}$$

$$\forall\, j \quad \sum_i \mathbf{x}_{ij} + \mathbf{r}_j \ge 1 \tag{4.4}$$

$$\mathbf{x}_{ij}, y_i, r_j \ge 0 \tag{4.5}$$

The dual of the above LP is:

$$\max \sum_j \alpha_j \tag{4.6}$$

$$\forall\, i \quad \sum_j \beta_{ij} \le f_i \tag{4.7}$$

$$\forall\, ij \quad \alpha_j \le c_{ij} + \beta_{ij} \tag{4.8}$$

$$\forall\, j \quad \alpha_j \le p_j \tag{4.9}$$

$$\alpha_j, \beta_{ij} \ge 0 \tag{4.10}$$

We use the following LP relaxation (LP2) for the k-median problem with penalties:

$$\min \sum_{ij} c_{ij} \cdot \mathbf{x}_{ij} + \sum_j p_j \cdot \mathbf{r}_j \tag{4.11}$$

$$\forall\, ij \quad \mathbf{x}_{ij} \le \mathbf{y}_i \tag{4.12}$$

$$\forall\, j \quad \sum_i \mathbf{x}_{ij} + \mathbf{r}_j \ge 1 \tag{4.13}$$

$$\sum_i \mathbf{y}_i \le k \tag{4.14}$$

$$\mathbf{x}_{ij}, y_i, r_j \ge 0 \tag{4.15}$$

The dual of the above LP is:

$$\max \sum_j \alpha_j - k \cdot \mathbf{z} \tag{4.16}$$

$$\forall\, i \quad \sum_j \beta_{ij} \le \mathbf{z} \tag{4.17}$$

$$\forall\, ij \quad \alpha_j \le c_{ij} + \beta_{ij} \tag{4.18}$$

$$\forall\, j \quad \alpha_j \le p_j \tag{4.19}$$

$$\alpha_j, \beta_{ij}, z \ge 0 \tag{4.20}$$

The LP relaxation (LP3) for the robust facility location problem is as follows. We use ℓ to denote the number of excluded outliers, can be set to be $n - p$.

$$\min \sum_i f_i \cdot \mathbf{y}_i + \sum_{ij} c_{ij} \cdot \mathbf{x}_{ij} \tag{4.21}$$

$$\forall\, ij \quad \mathbf{x}_{ij} \le \mathbf{y}_i \tag{4.22}$$

$$\forall\, j \quad \sum_i \mathbf{x}_{ij} + \mathbf{r}_j \ge 1 \tag{4.23}$$

$$\sum_j \mathbf{r}_j \le \ell \tag{4.24}$$

$$\mathbf{x}_{ij}, y_i, r_j \ge 0 \tag{4.25}$$

The dual of the above LP is:

$$\max \sum_j \alpha_j - \ell \cdot \mathbf{q} \tag{4.26}$$

$$\forall\, i \quad \sum_j \beta_{ij} \le f_i \tag{4.27}$$

$$\forall\, ij \quad \alpha_j \le c_{ij} + \beta_{ij} \tag{4.28}$$

$$\forall\, j \quad \alpha_j \le \mathbf{q} \tag{4.29}$$

$$\alpha_j, \beta_{ij} \ge 0 \tag{4.30}$$

The LP relaxation (LP4) for the robust k-median problem is:

$$\min \sum_{ij} c_{ij} \cdot \mathbf{x}_{ij} \tag{4.31}$$

$$\forall\ ij \quad \mathbf{x}_{ij} \leq \mathbf{y}_i \tag{4.32}$$

$$\forall\ j \quad \sum_i \mathbf{x}_{ij} + \mathbf{r}_j \geq 1 \tag{4.33}$$

$$\sum_i \mathbf{y}_i \leq k \tag{4.34}$$

$$\sum_j \mathbf{r}_j \leq \ell \tag{4.35}$$

$$\mathbf{x}_{ij}, y_i, r_j \geq 0 \tag{4.36}$$

The dual of the above LP is:

$$\max \sum_j \alpha_j - k \cdot \mathbf{z} - \ell \cdot \mathbf{q} \tag{4.37}$$

$$\forall\ i \quad \sum_j \beta_{ij} \leq \mathbf{z} \tag{4.38}$$

$$\forall\ ij \quad \alpha_j \leq c_{ij} + \beta_{ij} \tag{4.39}$$

$$\forall\ j \quad \alpha_j \leq \mathbf{q} \tag{4.40}$$

$$\alpha_j, \beta_{ij} \geq 0 \tag{4.41}$$

5 Primal dual algorithms

5.1 Facility location overview. Jain and Vazirani [19] gave a primal dual algorithm for the facility location problem. We first review their algorithm briefly as it will be an important building block in obtaining algorithms for the problems we consider.

The algorithm works in two phases. The first phase grows dual variables α_j and β_{ij} (initially 0). The variables α_j are grown uniformly. Suppose $\alpha_j > c_{ij}$, then β_{ij} is set to $\alpha_j - c_{ij}$; such an edge ij is called *saturated*. As this process continues, the following events occur:

1. $\sum_j \beta_{ij} = f_i$.
   In this case we say that facility i is *paid for*. For all j with $\beta_{ij} > 0$, α_j stops growing and i is said to be the *connecting witness* of j (provided j has not been assigned a connecting witness already). Let t_i be the time at which facility i gets paid for.

2. $\alpha_j = c_{ij}$ and i is paid for.
   In this case, α_j stops growing and i is said to be a connecting witness of j.

The first phase terminates when all demand points are assigned connecting witnesses. The second phase is a cleanup phase. We pick facility i that is paid for the earliest, delete facilities reachable from i by two saturated edges and repeat this process. A demand point j is said to be *directly connected* if there exists a facility i in the final solution such that $\beta_{ij} > 0$; j is assigned to i in this case. (Note that there can be at most one such facility i because of the cleanup step). On the other hand, if there is no facility i in the final solution such that $\beta_{ij} > 0$, then the demand point j is either assigned to its connecting witness or to the facility that caused the deletion of its connecting witness in the cleanup step. Such demand points are called *indirectly connected.*

Let C be the assignment cost of the solution produced by the algorithm, F be the facility cost and OPT be the cost of the optimal solution. Jain and Vazirani prove the following guarantee about their algorithm:

Lemma 5.1.

$$C + 3F \leq 3 \sum_j \alpha_j \leq 3OPT$$

We now show how the primal dual algorithm can be adapted to solve the penalty versions and the robust versions of the facility location and k-median problems.

5.2 Facility location with penalties. The dual of LP1 suggests a natural modification to the primal dual algorithm to adapt it for this problem. The constraint (4.9) puts an upper bound on the value of the dual variable α_j. Note that this is the only additional constraint compared to the dual for the facility location problem.

We modify the primal dual algorithm as follows: We grow the dual variable α_j as in the Jain Vazirani algorithm. If vertex j does not get a connecting witness by the time the value of α_j equals the penalty p_j, then we freeze the value of the dual variable α_j at p_j. In this case, we say that a *timeout* occurs for vertex j. We now run the rest of the algorithm with the value of α_j fixed at p_j. Note that later on in the algorithm, j could get a connecting witness. This happens if any facility i such that $\beta_{ij} > 0$ gets paid for. We also run the cleanup phase in an identical fashion as the original algorithm. Some of the vertices for which timeouts occurs are selected to be outliers as follows: If a timeout occurs for vertex j and j ends up being indirectly connected to a facility, then j is selected to be an outlier. All the other vertices are connected (either directly or indirectly) to the facilities assigned to them by the cleanup phase of the facility location algorithm.

Let C be the assignment cost of the solution produced by the algorithm, F be the facility cost of the solution, P be the total penalty for all the vertices selected as outliers, and OPT be the cost of the optimal solution. We can prove the following guarantee about the solution produced by the algorithm:

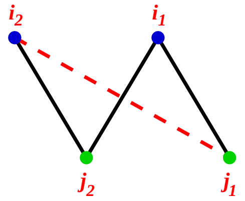

Figure 2: Indirectly connected demand point.

Lemma 5.2.

$$C + 3F + 3P \leq 3\sum_j \alpha_j \tag{5.42}$$

$$\leq 3OPT \tag{5.43}$$

Proof. (Sketch) The analysis proceeds along similar lines to the analysis of the primal dual algorithm for facility location. For a point j selected as an outlier, $\alpha_j = p_j$. The contribution to the LHS of (5.42) is $3p_j$ and the contribution to the RHS is $3\alpha_j$. We claim that the analysis for directly connected points and indirectly connected points goes through as in the original analysis. Note that neither of these contribute to the $3P$ term in the LHS of (5.42). The only properties of the algorithm used in the analysis are:

1. $\beta_{ij} > 0 \Rightarrow \alpha_j \leq t_i$.
2. i_1 is a connecting witness for $j_1 \Rightarrow \alpha_{j_1} \geq t_{i_1}$.

The first property holds for the modified algorithm. The second one is not necessarily true. However it holds if a timeout has not occurred for j_1. The only place we need the second property is in the analysis of an indirectly connected demand point (see Figure 5.2).

In this case it is certainly true that a timeout did not occur for j_1 (else j_1 would be selected to be an outlier). Hence the original analysis applies for both directly connected and indirectly connected demand points.

We thus obtain:

Theorem 5.1. *We can obtain a 3 approximation for the facility location problem with penalties in polynomial time.*

5.3 Robust facility location. We now describe a primal dual algorithm for the robust facility location problem, based on the relaxation LP3 (4.21) - (4.25). As stated, LP3 has an unbounded integrality gap. Since we use the LP to lower bound the cost of the optimal solution, we cannot hope to prove a bounded approximation guarantee using LP3. In order to fix this, we guess the most expensive facility in the optimal solution and run the algorithm on a modified instance where the only allowable facilities are those that are not more expensive than the guessed facility.

In fact, we guess the most expensive facility in the optimal solution (of cost f' say) and modify the instance by setting its cost to zero. For all facilities whose cost is greater than f', we set the facility cost to ∞. We run the primal dual facility location algorithm on this modified instance. However we terminate the execution of the algorithm before all vertices are connected to facilities. Initially all vertices are labeled outliers. As the algorithm proceeds, vertices are assigned *connecting witnesses* and they cease to be outliers. We stop the algorithm when the number of outliers is at most ℓ. In order to get exactly ℓ outliers, we examine the last step where the number of outliers first fell to a value $\leq \ell$. At this point, some facility i_f got paid for and a number of vertices got *connecting witnesses*. Of these, we select an arbitrary subset to be outliers so that the number of outliers is exactly ℓ. We decide which facilities to open by performing a cleanup step exactly as the cleanup step in the primal dual facility location algorithm. The facility i_f which got paid for at the termination of the algorithm may or may not get opened in the final solution.

We now analyze the algorithm. Suppose that in the optimal solution (of cost OPT) for the original robust facility location instance, the cost of the most expensive facility is f'. We separate the cost of the optimal solution (OPT) into the cost of the most expensive facility f' and the rest of the solution cost (OPT'). $OPT = OPT' + f'$. We focus on the case when the algorithm guesses f' correctly. The instance is modified by setting to ∞ the cost of all facilities originally more expensive than f'. Also the cost of the guessed most expensive facility is set to zero. Let S be the set of vertices that are connected to facilities, i.e. the set of all vertices excluding the ℓ outliers. The cost of the solution produced by the algorithm is broken up into the cost f_1 of the guessed most expensive facility, the cost f_2 of the facility i_f that got paid for at algorithm termination and the rest of the solution cost (the assignment cost C and the remaining facility cost F). Note that the facility i_f may or may not be selected to be opened in the final solution after the cleanup step is performed.

Lemma 5.3.

$$\sum_{j\in S} \alpha_j \leq OPT' \tag{5.44}$$

Proof. The original optimal solution is a feasible solution for the modified instance with cost $OPT' = OPT - f'$.

We will use the dual solution constructed by the algorithm to get a feasible solution for the dual of LP3 (4.26)-(4.30). Let t be the time at which the algorithm terminates, i.e., the time at which we have $\le \ell$ vertices remaining to be connected for the first time. We set the variable $\mathbf{q} = t$. Note that $\alpha_j \le t$, satisfying constraint (4.29). Constraints (4.27) and (4.28) are satisfied since these inequalities are maintained by the primal dual algorithm during its execution. The value of the dual solution is $\sum_j \alpha_j - \ell \cdot \mathbf{q} = \sum_{j \in S} \alpha_j$. The lemma follows from the fact that this is a lower bound on OPT',

Lemma 5.4.

$$C + 3F \quad \le \quad 3 \sum_{j \in S} \alpha_j \tag{5.45}$$

Proof. As in the proof of Jain and Vazirani, we charge the facility cost F and connection cost C to the dual variables α_j as follows. For vertex j *directly connected* to facility i, define $\alpha_j^f = \beta_{ij}$ (contribution to facility cost) and $\alpha_j^e = c_{ij}$ (contribution to connection cost). Then $\alpha_j = c_{ij} + \beta_{ij} = \alpha_j^e + \alpha_j^f$. For every open facility i that contributes to the facility cost F, $\sum_j \beta_{ij} = f_i$. The vertices j that have a positive contribution to $\sum_j \beta_{ij}$ are precisely the vertices that are *directly connected* to i. Note that all such vertices occur in S, i.e. are not selected as outliers. (The only possible exception is the case $i = i_f$, the facility that got paid for at the point of termination of the algorithm. Some of the vertices j for which $\beta_{ij} > 0$ may be selected as outliers. But notice that we exclude the cost of i_f from the facility cost F and account for it separately.) Thus, for every facility i that contributes to F, $f_i = \sum_j \alpha_j^f$ where the summation is over vertices j in S that are directly connected to i.

For every directly connected vertex j, the connection cost is α_j^e. Its contribution to the LHS of (5.45) is $\alpha_j^e + 3\alpha_j^f \le 3\alpha_j$. For every indirectly connected vertex j, the connection cost is at most $3\alpha_j$. Thus its contribution to the LHS is at most $3\alpha_j$.

Theorem 5.2. *For the robust facility location problem we can obtain a 3 approximation in polynomial time.*

Proof. Lemmas 5.4 and 5.3 imply that $C + 3F \le 3OPT'$. In addition to the facility costs included in F, the cost of the final robust facility location solution may include the cost f_1 of the guessed most expensive facility (recall that its cost is set to zero in the modified instance) as well as the cost f_2 of the facility that got paid for at the termination of the primal dual algorithm. Now $f_1 = f'$ and $f_2 \le f'$. The cost of the final solution is at most $C + F + f_1 + f_2 \le 3OPT' + 2f' \le 3OPT$.

5.4 k-median with penalties. Jain and Vazirani show how their primal dual approximation algorithm for facility location can be used to obtain an approximation algorithm for k-median, achieving an approximation ratio of 6. Charikar and Guha [4] improve the algorithm and to obtain a factor 4 approximation. Similarly, we can use the facility location with penalties algorithm to obtain an algorithm for k-median with penalties.

The basic idea is to take an instance of k-median with penalties, set all facility costs to z (where z is some parameter) and run the facility location with penalties algorithm on this instance. We do a binary search on z to find two values z_1 and z_2 very close such that for $z = z_1$, we obtain a solution with $\hat{k_1} \le k$ centers and for $z = z_2$, we obtain a solution with $\hat{k_2} \ge k$ centers. After this, we perform the greedy augmentation step introduced in [4]. This tries to add centers chosen in one solution to the other solution according to a certain rule.

We describe how the small solution (corresponding to z_1) is augmented. The augmentation of the large solution is done in a symmetric fashion other than the stopping condition.

1. Consider all nodes which are centers in the large solution and are not centers in the small solution. Let i be the current node.
2. If the small solution has k centers, stop.
3. If there exists a node i' which is chosen as a center in the current small solution (which resulted from a previous augmentation) and demand node j such that $\beta_{ij}(z_1) > 0, \beta_{ij}(z_2) > 0, \beta_{i'j}(z_1) > 0, \beta_{i',j}(z_2) > 0$, i cannot be added. Otherwise add node i to the small solution.
4. Repeat the above steps until the solution has k nodes or all nodes are considered.

The final solution to the k-median problem will be either one of the two solutions obtained after augmentation or a subset of centers in either of the solutions, under a suitable distribution.

We claim that a modified form of the analysis of [4] goes through for the k-median problem with penalties. The guarantee we obtain is as follows:

Theorem 5.3. *Let C be the assignment cost of the solution returned by the algorithm, P be the total penalty for the vertices selected as outliers and OPT be the cost of the optimal solution. Then,*

$$C + 4P \le 4OPT \tag{5.46}$$

5.5 Robust k-median. The gap example in Appendix A shows that we cannot expect to obtain a constant factor guarantee for the robust k-median problem by using the linear relaxation LP4 (4.31) - (4.36) as a lower bound. However, we can get a bi-criteria approximation guarantee using the previous result. We take a robust k-median instance, set penalties of vertices appropriately and use the solution returned by running the k-median with penalties algorithm on this instance. The penalties are chosen as follows: Suppose we know the value C^* of the optimal solution to the robust k-median problem with at most ℓ outliers. We set the penalties of vertices to be $C^*/\gamma\ell$ (γ is a tradeoff parameter that we can set). Since we do not know the exact value C^* of the optimal solution, we guess C^* to within $1+\epsilon$.

Theorem 5.4. *The algorithm returns a solution to the robust k-median problem with number of outliers at most $(1+\gamma)$ times the optimal solution and cost at most $4(1+1/\gamma)$ times the cost of the optimal solution (within $(1+\epsilon)$ factors).*

Proof. For ease of presentation, we will first assume that the penalties are set to exactly $C^*/\gamma\ell$, where C^* is the cost of the optimal solution to the robust k-median problem with ℓ outliers. For the instance of k-median with penalties, the value OPT of the optimal solution is at most

$$OPT \leq C^* + \ell \cdot \frac{C^*}{\gamma\ell} = C^*\left(1+\frac{1}{\gamma}\right).$$

Suppose the algorithm for the k-median problem with penalties returns a solution of cost C with ℓ' outliers. Then

$$\begin{aligned} C + 4\ell'\frac{C^*}{\gamma\ell} &\leq 4C^*\left(1+\frac{1}{\gamma}\right) \\ C &\leq 4C^*\left(1+\frac{1}{\gamma}\right) \\ \ell' &\leq \ell(1+\gamma) \end{aligned}$$

Since we guess the value of C^* to within a $(1+\epsilon)$ factor, the above guarantees on C and ℓ hold to within $(1+\epsilon)$ factors.

Acknowledgments

We would like to thank Sudipto Guha for useful discussions. In particular, Sudipto fixed a bug in the lower bound proof for the robust k-center problem.

References

[1] S. Arora. Polynomial time approximation schemes for Euclidean TSP and other geometric problems. In *Proc. 37th Annu. IEEE Sympos. Found. Comput. Sci.*, pages 2–11, 1996.

[2] S. Arora, P. Raghavan, and S. Rao. Approximation schemes for Euclidean k-median and related problems. In *Proc. 30th Annu. ACM Sympos. Theory Comput.*, pages 106–113, 1998.

[3] M. L. Balinski, "On finding integer solutions to linear programs", *Proc. of the IBM Scientific Computing Symposium on Combinatorial Problems*, pages 225–248, (1966).

[4] M. Charikar and S. Guha. Improved combinatorial algorithms for the facility location and k-median problems. In *Proc. 40th Annu. IEEE Sympos. Found. Comput. Sci.*, pages 378–388, 1999.

[5] M. Charikar, S. Guha, E. Tardos, and D. Shmoys. A constant factor approximation algorithm for the k-median problem. In *Proc. 31st Annu. ACM Sympos. Theory Comput.*, pages 1–10, 1999.

[6] F. Chudak. Improved approximation algorithms for uncapacitated facility location. In *Integer programming and combinatorial optimization*, pages 180–194, 1998.

[7] G. Cornuejols, G. L. Nemhauser and L. A. Wolsey, "The uncapacitated facility location problem", *Discrete Location Theory* (ed: Mirchandani and Francis), pages 119–171, (1990).

[8] T. Feder and D. H. Greene. Optimal algorithms for approximate clustering. In *Proc. 20th STOC*, pages 434–444, 1988.

[9] U. Feige. A threshold of $\ln n$ for approximating set cover. *J. ACM*, 45(4):634–652, 1998.

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

[11] D. Bienstock, M. Goemans, D. Simchi-Levi and D. Williamson. A Note on the Prize Collecting Traveling Salesman Problem. *Mathematical Programming*, 59, pages 413–420 (1993).

[12] M. Dyer and A. M. Frieze. A simple heuristic for the p-center problem. *Operations Research Letters*, 3:285–288, 1985.

[13] Sudipto Guha and Samir Khuller. Greedy strikes back: Improved facility location algorithms. In *Proc of the 9th Annual ACM-SIAM Symp. on Discrete Algorithms*, pages 649–657, 1998.

[14] Martha Hamilton. Loud and clear, a silent e. *The Washington Post*, April 23 2000.

[15] D. S. Hochbaum, "Heuristics for the fixed cost median problem", *Mathematical Programming*, 22: 148–162, (1982).

[16] D. Hochbaum. *Approximation Algorithms for NP-hard problems.* PWS Publishing, 1995.

[17] D. S. Hochbaum and D. B. Shmoys. A unified approach to approximation algorithms for scheduling problems: Practical and theoretical results. *J. ACM*, 33:533–550, 1986.

[18] A. Jain and R. Dubes. *Algorithms for Clustering Data.* Prentice Hall, Englewood Cliffs, NJ, 1988.

[19] K. Jain and V. Vazirani. Primal-dual approximation algorithms for metric facility location and k-median

problems. In *Proc. 40th Annu. IEEE Sympos. Found. Comput. Sci.*, pages 2–13, 1999.

[20] S. Khuller, A. Moss, and J. Naor. The budgeted maximum coverage problem. *IPL*, 70(1):39–45, 1999.

[21] M. Korupolu, C. G. Plaxton and R. Rajaraman, "Analysis of a local search heuristic for facility location problems", *9th ACM-SIAM Annual Symposium on Discrete Algorithms*, pages 1–10, (1998).

[22] A. A. Kuehn and M. J. Hamburger, "A heuristic program for locating warehouses", *Management Science*, 9:643–666, (1963).

[23] J.-H. Lin and J. S. Vitter, "Approximation algorithms for geometric median problems", *Information Processing Letters*, 44:245–249, (1992).

[24] A. S. Manne, "Plant location under economies-of-scale-decentralization and computation", *Management Science*, 11:213–235, (1964).

[25] J. Matoušek, D. Mount, and N. Netanyahu. Efficient randomized algorithms for the repeated median line estimator. In *Proc. 4th SODA*, pages 74–82, 1993.

[26] D. Mount, N. Netanyahu, and J. LeMoigne. Efficient algorithms for robust point pattern matching and applications to image registration. In *Proc. of the CESDIS Image Registration Workshop*, pages 247–256, 1997.

[27] D. Mount, N. Netanyahu, K. Romanik, R. Silverman, and A. Yu. A practical approximation algorithm for the lms line estimator. In *Proc. 8th SODA*, pages 473–482, 1997.

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

[29] J. Plesnik. A heuristic for the p-center problem in graphs. *Discrete Applied Mathematics*, 17:263–268, 1987.

[30] P. J. Rousseeuw and A. M. Leroy. *Robust Regression and Outlier Detection.* Wiley, New York, 1987.

[31] D. Shmoys, E. Tardos, and K. Aardal. Approximation algorithms for facility location problems. In *Proc. 29th Annu. ACM Sympos. Theory Comput.*, pages 265–274, 1997.

[32] J. F. Stollsteimer, "A working model for plant numbers and locations", *J. Farm Econom.*, 45: 631–645, (1963).

Appendix

A Gap example for the robust k-median LP relaxation

We give an example to show that a bicriteria result is inevitable if we use LP4 (4.31)-(4.36) as a lower bound on the value of the optimal solution to k-median with excluded outliers. Consider an instance consisting of $2n$ vertices at x, $2n$ vertices at y and n vertices at z. The distances between points are: $c_{xy} = 1, c_{xz} = \infty, c_{yz} = \infty$. We describe two solutions to the robust 2-median problem for this instance. The first solution S_1 consists of centers at x and y; the vertices at z are outliers. The cost of this solution is 0 and the number of outliers is n. The second solution S_2 consists of centers at x and z. The vertices at y are connected to the center at x. The cost of this solution is $2n$ and the number of outliers is 0. Consider the fractional solution $S = (1 - \frac{1}{n})S_1 + \frac{1}{n}S_2$. This is a feasible solution to LP4 with the number of outliers $\ell = n - 1$ and cost 2. However the optimal cost of an integral solution with at most $n-1$ outliers is $n+1$. (For example, this can be achieved by placing centers at x and z. $n + 1$ of the vertices at y are connected to the center at x and $n - 1$ of them are selected to be outliers). This implies that the integrality gap of LP4 is at least $(n + 1)/2$.