The Facility Location Problem With General Cost Functions

MohammadTaghi Hajiaghayi*, Mohammad Mahdian*, and Vahab S. Mirrokni*

Laboratory for Computer Science, Massachusetts Institute of Technology

Abstract. In this paper we define and study a generalized version of facility location problem in which facility cost functions depend on the number of clients assigned to the facility. There is an associated cost function for each facility that depends on the number of clients assigned to it. We focus on the case of concave facility cost functions, and present a greedy 1.861 approximation algorithm for this case. Then, we show a 1.52 approximation algorithm for this problem based on a reduction to the uncapacitated facility location problem. We will also consider generalizations and variants of the problem.
Keywords: Approximation algorithms, the facility location problem, facility costs, connection costs, concave (convex) functions.

1 Introduction

The facility location problem is a central problem in operation research. In this problem, we have a set of clients and a set of facilities, and we want to connect each client to a facility in a way that minimizes the total cost. There are two types of costs associated with a solution: the connection costs and the facility costs. The *connection cost* between each client j and facility i is a number c_{ij} that is given. We will have to pay this amount if we want to connect client j to facility i. Each facility i also has a *facility cost* f_i, that is the amount we have to pay if we decide to connect at least one client to it. The cost of a solution is the summation of the connection costs and the facility costs associated with the solution. We will generally assume that the connection costs obey the metric inequality.

The facility location problem is well-studied in the field of approximation algorithms, and a number of different approximation algorithms have been proposed for this problem using a variety of techniques [13, 16, 7, 8, 12, 9, 1–5, 9, 10, 15].

A natural generalization of the facility location problem is to allow the facility cost of a facility to be an arbitrary concave function of the number of cities connected to it. This question was first asked by T. Leighton [11] and was motivated by its applications in placing servers on the Internet. The reason it is assumed that the facility cost function is concave is the following principle of economics: As the number of clients increases, the cost per client will decrease, since they share some common expenses.

First we present a 1.861 approximation algorithm for the concave facility location problem. Our algorithm is a natural generalization of an algorithm by Mahdian et al. [12]. The

* Emails: {hajiagha, mahdian, mirrokni}@theory.lcs.mit.edu

analysis uses the techniques of dual-fitting and factor-revealing LPs introduced in [7]. Then, we show a reduction of this problem to the uncapacitated facility location problem which yields a 1.52 approximation algorithms whose running time is $O(n^2)$ worse than that of the greedy algorithm. We will also consider more general versions of the problem by relaxing the conditions of concavity for facility costs and metricity for connection costs.

This paper is organized as follows. In section 2, we give a formal definition of the problem, and observe that it can be reduced to the capacitated facility location problem with hard capacities. In section 3, we present a greedy 1.861 approximation algorithm for the concave facility location problem. In section 4, we prove the approximation factor 1.861 for our greedy algorithm and then show the reduction to the uncapacitated facility location problem. In section 5, we study generalizations and variants of the problem such as allowing the facility cost function to be slightly non-concave, facility location with convex cost functions, and relaxing the metric inequality for the connection costs.

2 The problem

The *generalized facility location* problem is defined as follows. We are given a set $\mathcal{F}$ of *facilities* (a.k.a servers); a set $\mathcal{D}$ of *clients* (a.k.a. cities or demands); a non-decreasing *facility cost function* $f_i : N \to N$ for every facility $i \in \mathcal{F}$, which specifies the cost of the facility as a function of the number of clients served by it; and finally *connection costs* c_{ij} between facility i and client j. We generally assume that the connection costs are *metric* (i.e., are symmetric and satisfy the triangle inequality), unless otherwise stated. The objective of the problem is to find an assignment ψ of *all* clients to facilities (i.e., $\psi : \mathcal{D} \mapsto \mathcal{F}$) with minimum total cost. The total cost is the sum of connection costs ($\sum_{j\in\mathcal{D}} c_{\psi(j),j}$) and facility costs ($\sum_{i\in\mathcal{F}} f_i(|\{j \in \mathcal{D} : \psi(j) = i\}|)$).

The focus of this paper is on the *concave facility location problem*, i.e., the generalized facility location problem with a guarantee that all facility cost functions are concave. Recall that a function $f : N \to N$ is called concave if and only if for each $x \geq 1$, $f(x+1)-f(x) \leq f(x)-f(x-1)$.

We observe that the concave facility location problem can be reduced to the *capacitated facility location with hard capacities*. In the capacitated facility location problem, each facility i also has a capacity u_i, which is the maximum number of clients that can be served by this facility. The problem has two variants; in the first one, the capacities are *hard* in that facility i can be opened at most once to serve at most u_i clients; in the second variant the capacities are *soft*; that is, facility i may be opened k times to serve up to ku_i clients at a cost of kf_i. It is easy to observe that both of these problems are special cases of the generalized facility location problem. The capacitated facility location problem with soft capacities can be reduced to the uncapacitated facility location problem [13], and the best known approximation algorithm for this problem is obtained via this reduction [13]. However, hard capacities are known to be problematic for techniques that rely on linear programs (such as LP-rounding and primal-dual algorithms). Recently, Pal et al. [14] used local search techniques to give the first constant factor approximation algorithm for this problem, achieving a factor of $9+\epsilon$. However, the running time of this algorithm is prohibitive for applications with large data sets. In the following, we show that the concave facility location problem can be reduced to the capacitated facility location problem with hard capacities.

Proposition 1. *Assume there is a polynomial time α-approximation algorithm for the capacitated facility location problem with hard capacities. Then the concave facility location problem can be approximated within a factor of α in polynomial time.*

Proof. For each facility $i \in \mathcal{F}$, we place n copies of this facility, where the j'th copy has opening cost $f_i(j)$ and capacity j. We use the α-approximation algorithm for the capacitated facility location problem to solve this instance. Since the facility cost function is concave, we can assume without loss of generality that at most one of these n copies is opened in this solution. Therefore, the solution to the capacitated facility location instance can be easily transformed to a solution for the concave facility location problem with the same total cost.

The above proposition together with the algorithm of Pal et al. [14] gives a $9+\epsilon$-approximation algorithm for the concave facility location problem.

3 The Greedy Algorithm

In this section, we present a greedy 1.861-approximation algorithm for the *concave facility location problem*. Our method is an extension of the method used by Mahdian et al. [12] and Jain et al. [8]. We start by the following definition from [8].

Definition. A *star* consists of one facility and several clients. For star S, consisting of clients $a_1, a_2, \ldots, a_k$ and facility p, the cost of S, denoted by $c(S)$ or c_S, is defined as $\sum_{i=1}^{k} c_{p,a_i} + f_p(k)$, i.e., the sum of the connection costs of clients to the facility p plus the facility cost function of p for k clients.

Suppose our algorithm finds a solution SOL of cost $cost(SOL)$ and values α_j for every client $j \in \mathcal{C}$ such that $\sum_{j\in\mathcal{C}} \alpha_j = cost(SOL)$ (one can think of α_j as the contribution of client j in the total cost). In addition, assume and there is a constant $\gamma \geq 1$ such that for every star S, $\sum_{j\in S\cap\mathcal{C}} \alpha_j \leq \gamma c_S$. We now consider an optimal solution OPT to the problem. Let i be a facility that is opened in OPT, D_i be the set of clients that are served by it, and S_i be the star consisting of i and D_i. By writing the inequalities $\sum_{j\in D_i} \alpha_j \leq \gamma c(S_i)$ for every i that is open in OPT, we obtain $cost(SOL) = \sum_{j\in\mathcal{C}} \alpha_i \leq \gamma \sum_{S\in \mathrm{OPT}} c_S = \gamma \cdot cost(OPT)$. Therefore if we can prove the statement $\sum_{j\in S\cap\mathcal{C}} \alpha_j \leq \gamma c_S$ for every star S, then γ is an upper bound on the approximation factor of the algorithm. This technique is called *dual-fitting*, and is explained in [7].

In the next section using the approach discussed above, we show that the following greedy algorithm is a 1.861 approximation algorithm for the concave facility location problem. In this algorithm, we use a notion of time, such that every event can be associated with the time at which it occurs. Also each client j has a budget from which it can offer some money to facilities; if j is unconnected and its budget is more than the cost of the connection to a facility i, it offers the extra budget to i (line 10).

Algorithm A: greedy algorithm for concave facility location

Input: Metric connection costs c_{ij} for each facility i and client j.
Concave facility cost functions $f_i : N \to N$ for each facility i.

Output: For each client j, a facility $\psi(j)$ to which j is assigned.
For each client j, contribution of client j to the total cost (α_j).

```
begin
1   let t = 0
2   for each facility i let level_i = 0
3   for each client j
4       let ψ(j) = null
5       let budget(j) = 0
6   end for
7   while there is an unconnected client increase time t
8       for each unconnected client j let budget(j) = t
9       for each client j and each facility i
10          if ψ(j) = null let offer(j, i) = max(budget(j) − c_ij, 0)
11          else let offer(j, i) = 0
12      end for
13      if there is a facility i and clients a_1, ···, a_k such that Σ_{j=1}^{k} offer(a_j, i) = f_i(level_i + k) − f_i(level_i) then
14          let level_i = k
15          for each 1 ≤ j ≤ k let ψ(a_j) = i
16      end if
17  end while
18  for each client j
19      let α_j = budget(j) (i.e., the time that j first gets connected)
20  end for
end
```

The above algorithm is similar to the greedy algorithm of Mahdian et al. [12]. The proof of the following lemma is clear from the algorithm and the discussion above. By an analysis very similar to the analysis of Mahdian et al., the running time of Algorithm A is at most $O(n^4)$.

Lemma 1. *The total cost of the solution found by Algorithm A is equal to the sum of α_j's.*

4 The approximation factor

In this section, we show that Algorithm A is a 1.861-approximation algorithm for the concave facility location problem. We prove this by showing that for each star S, the ratio of the sum of α_j's of all clients contained in S to the cost of S is at most 1.861. As in [12], our approach is to write various constraints imposed by the structure of the algorithm as linear inequalities, and find the maximum ratio of the sum of α_j's of all clients in a star to the cost of the star by solving a linear program called the *factor-revealing LP*. See [7] for a detailed explanation of this technique. Luckily, the factor-revealing LP that we get is the same as the one derived in [12]. This allows us to use their bound of 1.861 on the solution of the factor-revealing LP.

Consider a star S consisting of a facility p and k clients numbered 1 through k. Let d_j denote the connection cost between facility p and client j, and α_j denote the share of j of the total expenses (see the definition of α in the algorithm). The cost of the star is $f_p(k) + \sum_{i=1}^{k} d_i$. For simplicity, we set $f = f_p(k)$. Without loss of generality, we assume $\alpha_1 \leq \alpha_2 \leq \cdots \leq \alpha_k$.

Lemma 2. *For every $i = 1, \ldots, k$, we have $\sum_{j=i}^{k} \max(\alpha_i - d_j, 0) \leq f$.*

Proof. Consider the time just before i gets connected for the first time, i.e., when $t = \alpha_i - \epsilon$ for very small ϵ. We call this the *critical time* for the client i. At this time, the amount client j offers to facility f is $\max(t - d_j, 0)$ for $j \geq i$. From the algorithm, the total offer of clients $1, \ldots, k$ to facility p can never exceed the cost of facility p for serving k clients, since otherwise all these clients would have been assigned to the facility p at an earlier time. This completes the proof of the lemma.

So far, we have not used the triangle inequality of connection costs and concavity of facility cost functions. We use these assumptions in the next lemma.

Lemma 3. *For all clients i and j such that $j < i$, we have $\alpha_i \leq \alpha_j + d_i + d_j$.*

Proof. Consider the critical time for client i ($t = \alpha_i - \epsilon$). Let p' be the facility that j is connected to at this time. Also, let l denote the value of $\text{level}_{p'}$ right after client j gets connected to p'. By concavity of f, at any time after this, the cost of serving a new client by p' is at most $f(l) - f(l-1)$. Therefore, we must have $t \leq c_{p'i} + f_{p'}(l) - f_{p'}(l-1)$; since otherwise i should have been connected at a time earlier than t to p'. Therefore,

$$\alpha_i \leq c_{p'i} + f_{p'}(l) - f_{p'}(l-1). \tag{1}$$

Now, consider the critical time for j, i.e., just before it gets connected to p', and let q denote the value of $\text{level}_{p'}$ at this moment. This means that there are $l - q$ new clients $b_1, \ldots, b_{l-q}$ that are connected to p' at time α_j. For simplicity, let b_{l-q} be j. First, notice that at the critical time for j we have $\sum_{i=1}^{l-q} \text{offer}(b_i, p') = f_{p'}(l) - f_{p'}(q)$. Also, we must have $\sum_{i=1}^{l-q-1} \text{offer}(b_i, p') = f_{p'}(l-1) - f_{p'}(q)$, since otherwise the clients $b_1, \ldots, b_{l-q-1}$ will be connected to p' at an earlier time. Combining the above inequalities, we obtain $\text{offer}(j, p') \geq f_{p'}(l) - f_{p'}(l-1)$, which implies

$$\alpha_j - c_{p',j} \geq f_{p'}(l) - f_{p'}(l-1). \tag{2}$$

Inequalities 1 and 2 imply $\alpha_i \leq c_{p',i} + \alpha_j - c_{p',j}$. By triangle inequality, $c_{p',i} \leq c_{p',j} + d_i + d_j$, and therefore $\alpha_i \leq \alpha_j + d_i + d_j$.

The above lemmas imply that the maximum ratio of $\sum \alpha_i$ to the cost of S can be bounded by the solution of the following maximization program.

$$
\begin{aligned}
\text{maximize} \quad & \frac{\sum_{i=1}^{k} \alpha_i}{f + \sum_{i=1}^{k} d_i} && (3)\\
\text{such that} \quad & \forall\, 1 \le i < k: \;\; \alpha_i \le \alpha_{i+1} \\
& \forall\, 1 \le j < i \le k: \;\; \alpha_i \le \alpha_j + d_i + d_j \\
& \forall\, 1 \le i \le k: \;\; \sum_{j=i}^{k} \max(\alpha_i - d_j, 0) \le f \\
& \forall\, 1 \le j \le i \le k: \;\; \alpha_j, d_j, f \ge 0
\end{aligned}
$$

Therefore, as in [12], if we let γ be $\sup_k\{z_k\}$, where z_k is the solution of the above maximization program, then Algorithm A is a γ-approximation algorithm for the concave facility location problem. This linear program is the same as the one derived in [12], and therefore we can use the following result of [12].

Lemma 4. *For every k, the solution of the maximization program 3 is less than 1.861.*

This, together with the above discussion implies the following.

Theorem 1. *Algorithm A is a 1.861-approximation algorithm for the concave facility location problem.*

It is worth mentioning that we can obtain a better approximation for concave facility location, but its running time is $O(n^2)$ factor worse than the running time of the above algorithm, since the running of Algorithm A is at most $O(n^4)$ and the running time of the new algorithm is $O(n^6)$ as we will see in the next theorem. To this end, we show a reduction from the concave facility location problem to the uncapacitated facility location problem which yields a 1.52-approximation algorithm. (This reduction was pointed out from SODA anonymous referee and was introduced independently by Jiawei Zhang).

Theorem 2. *There exists a 1.52-approximation algorithm for the concave facility location problem.*

Proof. The main idea is that in our problem any concave function $f(x)$ can be represented by $min_k\{L_k = (f(k) - f(k-1))x + kf(k-1) - (k-1)f(k)\}$, $1 \le k \le n$. Here L_k is the line passing through points $(k-1, f(k-1))$ and $(k, f(k))$. Now, we take each facility i with concave cost function f_i, and replace it by multiple facilities $i_1, i_2, \cdots, i_n$ such that facility i_k has opening cost $kf_i(k-1) - (k-1)f_i(k)$. In addition, each unit of demand routed to this location costs $f_i(k) - f_i(k-1)$ extra at the facility. This cost $f_i(k) - f_i(k-1)$ can be added to the connection costs still keeping them metric. For this facility location problem, Mahdian et al. [13] give a 1.52 approximation. The the running time of 1.52 approximation algorithm is $O(n^3)$[13], thus the running of this algorithm is $O(n^6)$.

5 Generalizations

In this section, we consider more general variants of the problem. So far, we have shown approximation algorithms for the concave facility location problem. In this section, we consider the generalized facility location problem with more general facility cost functions. First, we consider the problem with non-metric connection costs.

Theorem 3. *The concave facility location problem has a* $\ln n$*-approximation algorithm when we have arbitrary (not necessary metric) connection costs.*

Proof. We reduce the problem to set cover. For each set $S = \{d_1, \cdots, d_k\}$ of clients and a facility i, we implicitly consider a set $S' = S$ in a set cover instance with cost $f_i(k) + c_{id_1} + \cdots + c_{id_k}$. Here our goal is to cover all clients. We observe that in an optimal solution for the instance of set cover, each client will be picked by exactly one set (otherwise we can decrease the total cost by removing the client from all except one). Since for a concave function f, $f(i) + f(j) \geq f(i+j)$, in an optimal solution we pick at most one set corresponding to each facility i. The only problem here is that the number of sets in the instance is exponential, however one can show that selecting the most efficient set in each step of the greedy algorithm for set cover can be done in polynomial time and thus the overall running time of the algorithm is polynomial (see the algorithm for set cover in [8]). The greedy algorithm for set cover gives us a $\ln n$-approximation where n is the number of clients. □

Now, we give a polynomial time algorithm for the generalized facility location problem with convex facility cost functions. Recall that a function f is called *convex* if for every $x \geq 1$, $f(x+1) - f(x) \geq f(x) - f(x-1)$.

Theorem 4. *The generalized facility location problem with convex facility cost functions can be solved in polynomial time. (Here for each facility cost function* f*, we assume* $f(0) = 0$*.)*

Proof. First we reduce the problem to the capacitated facility location problem with unit hard capacities. Next, we show how this problem can be solved in polynomial time. For each facility $i \in \mathcal{F}$, we place n copies of unit-capacity facilities where f_i^j, the opening cost of the jth copy of facility $i \in \mathcal{F}$, is $f_i(j+1) - f_i(j)$, $0 \leq j \leq n-1$. The correctness of the reduction follows from the fact that if we use a facility f_i^j, we should use all facilities f_i^k, $k \leq j$, since f_i is convex. We now solve the capacitated facility location problem with unit capacities using minimum weighted matching algorithms on bipartite graphs. We construct a bipartite graph $G = (X \cup Y, E)$ as follows. For each client j, we place a vertex in the set X, for each facility i, we place a vertex in the set Y, and finally we place an edge $\{j, i\}$ in E between a client j and a facility i of weight $c_{ij} + f_i$. We can easily observe that by solving minimum weighted matching [17] for G, one can find an optimum assignment for the original problem.

The problem of finding a constant factor approximation algorithm for general facility cost functions is open. However, it is easy to observe that if each facility cost function f_i can be approximated by a concave function g_i, i.e., there is a function g_i such that for every x, $g_i(x)/c \leq f_i(x) \leq g_i(x)$, then applying Algorithm A on an instance with facility cost functions g_i gives us a $1.52c$-approximate solution for the original problem. If for function f, $f(x+1) - f(x) \leq c(f(x) - f(x-1))$ for all x, then by proving a relaxed version of the inequality in Lemma 3 and solving the corresponding factor-revealing LP, one can obtain the approximation factor of the algorithm, one can prove that in the Algorithm A of [6], $\alpha_i \leq \alpha_j + d_i + d_j + (c-1)r_{ji}$ (for r_{ji} see notations in [8], [6]). Specifically, for $c = 2$ it yields approximation factors 1.94 respectively [6].

6 Conclusions and Open problems

It is easy to see that the facility location problem (and therefore the generalized facility location problem) is NP-hard to approximate within a factor less than $O(\ln n)$ if the connection costs are not metric. Also, as mentioned in Theorem 3, the classical set cover algorithm can approximate the non-metric concave facility location problem within a factor of $O(\ln n)$. However, this reduction does not work for the non-metric *generalized* facility location problem. Obtaining a $\ln n$-approximation algorithm for this problem, where n is the number of clients, is an open problem. Another interesting open problem is obtaining a constant factor approximation for the generalized facility location problem with metric connection costs.

Acknowledgments. We would like to thank Tom Leighton, Rajmohan Rajaraman and Ravi Sundaram for introducing the problem. We especially thank Jiawei Zhang for his comments on the earlier draft of this paper.

References

1. V. Arya, N. Garg, R. Khandekar, A. Meyerson, K. Munagala, and V. Pandit. Local search heuristics for k-median and facility location problems. In *Proceedings of 33rd ACM Symposium on Theory of Computing*, 2001.
2. M. Charikar and S. Guha. Improved combinatorial algorithms for facility location and k-median problems. In *Proceedings of the 40th Annual IEEE Symposium on Foundations of Computer Science*, pages 378–388, October 1999.
3. F.A. Chudak. Improved approximation algorithms for uncapacited facility location. In R.E. Bixby, E.A. Boyd, and R.Z. Ríos-Mercado, editors, *Integer Programming and Combinatorial Optimization*, volume 1412 of *Lecture Notes in Computer Science*, pages 180–194. Springer, Berlin, 1998.
4. F.A. Chudak and D. Shmoys. Improved approximation algorithms for the uncapacitated facility location problem. unpublished manuscript, 1998.
5. S. Guha and S. Khuller. Greedy strikes back: Improved facility location algorithms. *Journal of Algorithms*, 31:228–248, 1999.
6. MohammadTaghi Hajiaghayi, Mohammad Mahdian, and Vahab S. Mirrokni. The facility location problem with concave cost functions. Technical Report MIT-LCS-TR-864, Massachusetts Institute of Technology, September 2002.
7. K. Jain, M. Mahdian, E. Markakis, A. Saberi, and V.V. Vazirani. Approximation algorithms for facility location via dual fitting with factor-revealing lp. unpublished.

8. K. Jain, M. Mahdian, and A. Saberi. A new greedy approach for facility location problem. In *Proceedings of 34rd ACM Symposium on Theory of Computing*, 2002.
9. K. Jain and V.V. Vazirani. Primal-dual approximation algorithms for metric facility location and k-median problems. In *Proceedings of the 40th Annual IEEE Symposium on Foundations of Computer Science*, pages 2–13, October 1999.
10. M.R. Korupolu, C.G. Plaxton, and R. Rajaraman. Analysis of a local search heuristic for facility location problems. In *Proceedings of the 9th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1–10, January 1998.
11. T. Leighton. Internet Research seminars at MIT, February 2002.
12. M. Mahdian, E. Markakis, A. Saberi, and V.V. Vazirani. A greedy facility location algorithm analyzed using dual fitting. In *Proceedings of 5th International Workshop on Randomization and Approximation Techniques in Computer Science*, volume 2129 of *Lecture Notes in Computer Science*, pages 127–137. Springer-Verlag, 2001.
13. M. Mahdian, Y. Ye, and J. Zhang. Improved approximation algorithms for metric facility location problems. to appear in APPROX 2002, 2002.
14. M. Pal, E. Tardos, and T. Wexler. Facility location with hard capacities. *Proceedings of the 42nd Annual IEEE Symposium on the Foundations of Computer Science*, 2001.
15. D.B. Shmoys, E. Tardos, and K.I. Aardal. Approximation algorithms for facility location problems. In *Proceedings of the 29th Annual ACM Symposium on Theory of Computing*, pages 265–274, 1997.
16. M. Sviridenko. An 1.582-approximation algorithm for the metric uncapacitated facility location problem. Ninth Conference on Integer Programming and Combinatorial Optimization, 2002.
17. Douglas B. West. *Introduction to Graph Theory.* Prentice Hall Inc., Upper Saddle River, NJ, 1996.