On Approximating Facility Location and Related Problems

Mohammad Mahdian* Yinyu Ye† Jiawei Zhang ‡

Abstract

Variants of the facility location problems have been studied extensively in the operations research and management science literatures and have received considerable attention in the area of approximation algorithms. The result of this paper is twofold. First, we give an almost linear running time version of the authors' [20] 1.52-approxmation algorithm for the uncapacitated facility location problem (UFLP). The improvement is obtained by a new analysis of the algorithm, and by a recent result of [28]. Therefore, the 1.52-approximation algorithm not only possesses the currently best approximation ratio but also the fastest running time. Secondly, we define the concept of the *trade-off approximate reduction*. Then, we use the reduction and the result of UFLP to develop improved approximation ratios for several other location problems. In particular, we show that the soft-capacitated facility location problem can be approximated by a factor of 2, and this ratio is tight in the sense that a natural linear programming relaxation has an integral gap 2.

1 Introduction

Variants of the facility location problems have been studied extensively in the operations research and management science literatures and have received considerable attention in the area of approximation algorithms [24]. The metric uncapacitated facility location problem (UFLP) is one in which we are given a set $\mathcal{F}$ of *facilities* and a set $\mathcal{C}$ of *cities*, the cost f_i for opening facility $i \in \mathcal{F}$, and connection cost c_{ij} for connecting client j to facility i. The objective is to open a subset of the facilities in $\mathcal{F}$, and connect each city to an open facility so that the total cost is minimized. We assume that the connection costs are metric, meaning that they are symmetric and satisfy the triangle inequality.

Since the first constant factor approximation algorithm due to Shmoys, Tardos and Aardal [25], a large number of approximation algorithm have been proposed for UFLP [26, 14, 15, 19, 27, 16, 1, 3, 4, 5, 10, 16, 18], and the current best known approximation factor is 1.52 given by Mahdian, Ye and Zhang [20]. Regarding hardness results, Guha and Khuller [10] proved that it is impossible to get an approximation guarantee of 1.463 for UFLP, unless $\mathbf{NP} \subseteq \mathrm{DTIME}[n^{O(\log\log n)}]$.

The growing interests in UFLP rely on not only its applications in a large number of settings [7], but also the fact that UFLP is one of the most basic models among discrete location problems.

*Laboratory for Computer Science, MIT, Cambridge, MA 02139, USA. E-mail: mahdian@mit.edu.

†Department of Management Science and Engineering, School of Engineering, Stanford University, Stanford, CA 94305, USA. E-mail: yinyu-ye@stanford.edu. Research supported in part by NSF grant DMI-0231600.

‡Department of Management Science and Engineering, School of Engineering, Stanford University, Stanford, CA 94305, USA. E-mail: jiazhang@stanford.edu. Research supported in part by NSF grant DMI-0231600.

The insights gained in dealing with UFLP may also apply to more complicated location models, and in many cases the latter can be reduced directly to UFLP.

The goal of this paper is twofold. First, we show that UFLP can be approximated by a factor of 1.52 in $O(m \ln(n))$ times where $m := |\mathcal{F}| \cdot |\mathcal{C}|$ and $n := |\mathcal{F}| + |\mathcal{C}|$. This improves the previous running time $O(n^3)$ of the 1.52-approximation algorithm of Mahdian et al [20]. The algorithm of [20] consists of two phases, where the first phase is the greedy algorithm of Jain, Mahdian and Saberi [15], and the second phase is the greedy augmentation procedure of Guha and khuller [10] (also see Charikar and Guha [3]). Thorup [28] recently shows that the greedy algorithm of [15] can be implemented in $O(m \ln(n))$ times. Based on his analysis, we show that the first phase of the 1.52 algorithm can be also implemented in $O(m \ln(n))$ times. More interestingly, we give a new analysis or interpretation of the greedy augmentation procedure of [10, 3] and show that it can be done in $O(m)$ times without a negative effect on the overall approximation factor of 1.52. Thus, the 1.52 algorithm can be implemented in $O(m \ln(n))$ times; and the algorithm not only possesses the currently best approximation ratio, but also runs in almost linear time—currently the fastest. Furthermore, we show that the approximation ratio is given by the optimal value of a single linear program, the so-called factor revealing LP. Therefore, by constructing the worst-case LP solution, it is possible to give a tight example for the 1.52-approximation algorithm.

The second goal is to show how the 1.52 approximation result for UFLP can be extended for approximating many facility location variants who are special cases of the *generalized facility location problem* (GFLP). As in UFLP, in GFLP we are given a set $\mathcal{F}$ of facilities, a set $\mathcal{D}$ of clients, and metric connection cost c_{ij} for connecting client j to facility i. The difference is in the facility cost f_i to open $i \in \mathcal{F}$: it is a function of k, $f_i(k)$, where k the number of clients are assigned to i and $f(0) = 0$. The objective again is to connect each city to a facility such that the total connection and facility costs is minimized. This problem was defined and studied for non-metric connection costs in [13]. The special case, where $f_i(k)$ is a constant for $k \geq 1$, is exactly the UFLP. In the capacitated facility location problem with soft capacities [25, 18, 16, 20], we have $f_i(k) = a_i \lceil k/u_i \rceil$ for constants a_i and u_i; in the capacitated facility location problem with hard capacities [25, 18, 21] $f_i(k) = a_i$ for $1 \leq k \leq u_i$ and $f_i(k) = \infty$ for $k > u_i$; in load-balanced facility location problem [17, 11] $f_i(k) = \infty$ for $1 \leq k < l_i$ and $f_i(k) = a_i$ for $k \geq l_i$; and in the concave facility location problem [13] $f_i(k)$ is a general concave function for $k \geq 1$.

The approximation algorithms for UFLP have been serving as the key to design algorithms for many special cases of GFLP or related problems by reducing the latter to UFLP. Motivated by the idea of trade-off approximation for UFLP, we define the concept of the *trade-off approximate reduction*, and use it to show improved approximation ratios for the soft-capacitated facility location problem, a generalization of this problem, and a couple of other facility location problems. Under the framework of the trade-off approximate reduction, the improvements for some of these problems are straightforward.

In particular, we show that the soft-capacitated facility location problem (SFLP) can be approximated by a factor of 2, and the ratio is tight in the sense that the linear programming relaxation of this problem has an integrated gap of 2. In the SFLP, there is an upper bound u_i on the number of clients that can be served by facility i. However, we are allowed to open multiple copies of this facility. If r copies of facility i are opened, then the total cost for opening the r copies will be $r \cdot f_i$. In other words, if k clients have been assigned to i, then the total cost of this facility will be $\lceil \frac{k}{u_i} \rceil f_i$. This problem is also known as facility location problem with integer decision variables in the operations research literature, see [2]. Chudak and Shmoys [6] gave a 3-approximation algorithm for SFLP with uniform capacities using LP rounding, i.e., $u_i = u$ for all $i \in \mathcal{F}$. For non-uniform capacities, Jain and Vazirani [16] showed how to reduce this problem to UFLP, and by solving the UFLP through a primal-dual algorithm, they obtained a 4-approximation. A local search algorithm proposed by Arya et al [1] had an approximation ratio 3.72. Charikar and Guha [3] proved that any ρ-approximation algorithm for UFLP implies a 2ρ-approximation for SFLP, thus a 3.04-approximation algorithm followed from the result of [20]. Following the same approach of [16], Jain, Mahdian and Saberi showed that SFLP could be approximated by a factor of 3 using a greedy algorithm. This result was further improved by [20] which generalized the reduction of [16] and [3] and established a 2.89-approximation for SFLP. (For related soft-capacitated problems, see [8] and [9].)

The rest of this paper is organized as follows: In Section 2 we present the necessary definitions and notations. Then, we present a new analysis on the MYZ algorithm and show how it can be implemented in $O(m \ln(n))$ times in Section 3. In Section 4 we present a lemma on the approximability of the linear-cost facility location problem. In Section 5 we define the concept of reduction between facility location problems, which together with the lemma proved in Section 4, are used to approximate the soft capacitated facility location problem and a generalization of it. Finally, in Section 6 we show how similar reductions can be used to obtain improved approximation algorithms for other well-studied facility location problems.

2 Definitions and Notations

Metric generalized facility location problem (GFLP): Let G be a bipartite graph with bipartition $(\mathcal{F}, \mathcal{C})$ and edge set $\mathcal{E}$, where $\mathcal{F}$ is the set of facilities and $\mathcal{C}$ is the set of cities. Suppose also that $|\mathcal{C}| = n_c$ and $|\mathcal{F}| = n_f$. Thus, the total number of vertices in the graph is $n = n_c + n_f$ and the total number of edges is $m = n_c \times n_f$. Let $f_i(k)$ be the cost of opening facility i as a function of the number of cities that it serves in a solution, and c_{ij} be the cost of connecting city j to facility i. The connection costs are metric. A solution to the problem is a function $\phi : \mathcal{C} \to \mathcal{F}$ assigning each city to a facility. The *facility cost* $f(\phi)$ of the solution ϕ is defined as $\sum_{i \in \mathcal{F}} f_i(|\{j : \phi(j) = i\}|)$, i.e., the total cost for opening facilities. The *connection cost* (a.k.a. service cost) $c(\phi)$ of ϕ is $\sum_{j \in \mathcal{C}} c_{\phi(j),j}$, i.e., the total cost of opening each city to its assigned facility. The objective is to find a solution ϕ that minimizes the sum $f(\phi) + c(\phi)$.

Metric uncapacitated facility location problem (UFLP) is a special case of GFLP in which all facility costs are of the following form: for each $i \in \mathcal{F}$, $f_i(k) = 0$ if $k = 0$, and $f_i(k) = f_i$ if $k > 0$, where f_i is a constant.

The 1.52-approximation algorithm of Mahdian, Ye, and Zhang [20] is built upon an earlier 1.61-approximation algorithm of Jain, Mahdian, and Saberi [15]. We denote these two algorithms by the MYZ and the JMS algorithms, respectively. The analyses of both of these algorithms have the feature that allow the approximation factor for the facility cost to be different from the approximation factor for the connection cost (In fact, there is a tradeoff between the approximation factor for facility costs and the approximation factor for connection costs). The following definition captures this notion.

Definition 1 *An algorithm is called a (γ_f, γ_c)-approximation algorithm for GFLP, if for every instance $\mathcal{I}$ of GFLP, and for every solution SOL for $\mathcal{I}$ with facility cost F_{SOL} and connection cost C_{SOL}, the cost of the solution found by the algorithm is at most $\gamma_f F_{SOL} + \gamma_c C_{SOL}$.*

Recall the following theorem of Jain et al. [15] on the approximation factor of the JMS algorithm. We will use this theorem in the next section to give a simpler analysis of the MYZ algorithm.

Theorem A [15]. *Let $\gamma_f \geq 1$ be fixed and $\gamma_c := \sup_k\{z_k\}$, where z_k is the solution of the following optimization program (which we call the factor-revealing LP).*

$$
\begin{aligned}
&\text{maximize} && \frac{\sum_{i=1}^{k} \alpha_i - \gamma_f f}{\sum_{i=1}^{k} d_i} && (1)\\
&\text{subject to} && \forall\, 1 \leq i < k: \ \alpha_i \leq \alpha_{i+1} && (2)\\
& && \forall\, 1 \leq j < i < k: \ r_{j,i} \geq r_{j,i+1} && (3)\\
& && \forall\, 1 \leq j < i \leq k: \ \alpha_i \leq r_{j,i} + d_i + d_j && (4)\\
& && \forall\, 1 \leq i \leq k: \ \sum_{j=1}^{i-1} \max(r_{j,i} - d_j, 0) + \sum_{j=i}^{k} \max(\alpha_i - d_j, 0) \leq f && (5)\\
& && \forall\, 1 \leq j \leq i \leq k: \ \alpha_j, d_j, f, r_{j,i} \geq 0 && (6)
\end{aligned}
$$

Then the JMS algorithm is a (γ_f, γ_c)-approximation algorithm for the UFLP.

3 Uncapacitated Facility Location Problem

In this section, we present a new analysis of the 1.52-approximation algorithm of Mahdian, Ye, and Zhang [20] for the UFLP. The analysis of the MYZ algorithm in [20] is based on combining a result of Jain et al. [15] (which is proved using factor-revealing LPs) with an analysis of a greedy augmentation procedure of Charikar et al. [3]. Here, we analyze the MYZ algorithm using a single factor-revealing LP. This gives us a new perspective on the MYZ algorithm. As a corollary, we use a recent result of Thorup [28] that the JMS possesses an $O(m \ln(n))$ running time to improve the running time of the MYZ algorithm as well.

We begin by sketching the MYZ algorithm. The algorithm consists of two phases. In the first phase, we scale up the facility costs in the instance by a factor of δ (which will be fixed later), and then run the JMS algorithm (see [15] for a description) on the modified instance. In addition to find a solution for the scaled instance, the JMS algorithm outputs the *share* of each city of the total cost of the solution. Let α_j denote the share of city j of the total cost (Therefore the total cost of the solution is $\sum_j \alpha_j$). The main step in the analysis of the JMS algorithm is to prove that for any collection $\mathcal{S}$ of one facility A with opening cost δf (f in the original instance) and k cities with connection costs $d_1, \ldots, d_k$ to A and shares $\alpha_1, \ldots, \alpha_k$ of the total cost, the values δf, d_j's, α_j's and $r_{j,i}$'s (whose definition is omitted here, since we don't need it) satisfy the inequalities (2)-(6), except that the inequality (5) is replaced by

$$\forall\, 1 \leq i \leq k: \ \sum_{j=1}^{i-1} \max(r_{j,i} - d_j, 0) + \sum_{j=i}^{k} \max(\alpha_i - d_j, 0) \leq \delta f \tag{7}$$

As a result, using a *dual fitting* argument, they prove Theorem A. In [28], Thorup proposed a $O(m\ln(n))$ implementation of the JMS algorithm, which has an approximation ratio $1.61+\epsilon$ for any constant ϵ. In Appendix A, we show that his algorithm has an performance guarantee of $(\gamma_f+\epsilon, \gamma_c+\epsilon)$, where (γ_f, γ_c) are given by LP (1).

In the second phase of the MYZ algorithm we reduce the scaling factor δ continuously, until it gets to 1. If at any point during this process a facility could be opened without increasing the total cost (i.e., if the opening cost of the facility equals the total amount that cities can save by switching their "service provider" to that facility), then we open the facility and connect each city to its closest open facility. Of course, this phase cannot be implemented as it is described.

However, the second phase of the MYZ algorithm is equivalent to a greedy augmentation procedure of [10, 3], and, in fact, a lemma from [3] is used in [20] in order to analyze the second phase. Here we analyze the phase differently. First, we modify the second phase of the algorithm as follows: Instead of decreasing the scaling factor continuously from δ to 1, we decrease it discretely in L steps where L is a constant. Let δ_i denote the value of the scaling factor in the i'th step. Therefore, $\delta=\delta_1>\delta_2>\ldots>\delta_L=1$. We will fix the value of of δ_i's later. After decreasing the scaling factor from δ_{i-1} to δ_i, we consider facilities in an *arbitrary* order, and open those that can be opened without increasing the total cost. Note that the opening process can be completed in $m=n_c\times n_f$ running times for each δ_i. Therefore, the whole second phase can be completed in $Lm=O(m)$ running times. This is in contrast to the greedy augmentation procedure which may need $O(m\cdot n)$ running times. We denote this algorithm now by MYZ$_L$. Clearly, if L is sufficiently large, the algorithm MYZ$_L$ performs exactly the same as the MYZ algorithm.

In order to analyze the above algorithm, we need to add extra variables and inequalities to the inequalities (2), (3), (4), (7) and (6). Consider a collection $\mathcal{S}$ of one facility and k cities with variables $f, d_j, \alpha_j, r_{j,i}$ as defined above and in [15]. Let $r_{j,k+i}$ denote the connection cost that city j in $\mathcal{S}$ pays after we change the scaling factor to δ_i and process all facilities as described above (Thus, $r_{j,k+1}$ is just the connection cost of city j after the first phase). Therefore, by the description of the algorithm, we have

$$\forall\, 1\le i\le L:\ \sum_{j=1}^{k}\max(r_{j,k+i}-d_j,0)\le\delta_i f, \tag{8}$$

since otherwise we could open the facility in $\mathcal{S}$ and decrease the total cost.

Now, we compute the share of the city j of the total cost of the solution that the MYZ$_L$ algorithm finds. In the first phase of the algorithm, the share of city j of the total cost is α_j. Of this amount, $r_{j,k+1}$ is spent on the connection cost, and $\alpha_j-r_{j,k+1}$ is spent on the facility costs. However, since the facility costs are scaled up by a factor of δ in the first phase, therefore the share of city j of *facility costs* in the original instance is equal to $(\alpha_j-r_{j,k+1})/\delta$. After we reduce the scaling factor from δ_i to δ_{i+1} $(i=1,\ldots,L-1)$, the connection cost of city j is reduced from $r_{j,k+i}$ to $r_{j,k+i+1}$. Therefore, in this step, the share of city j of the facility costs is $r_{j,k+i}-r_{j,k+i+1}$ with respect to the scaled instance, or $(r_{j,k+i}-r_{j,k+i+1})/\delta_{i+1}$ with respect to the original instance. Thus, at the end of the algorithm, the total share of city j of facility costs is

$$\frac{\alpha_j-r_{j,k+1}}{\delta}+\sum_{i=1}^{L-1}\frac{r_{j,k+i}-r_{j,k+i+1}}{\delta_{i+1}}.$$

We also know that the final amount that city j pays for the connection cost is $r_{j,k+L}$. Therefore, the share of the facility j of the total cost of the solution is:

$$\frac{\alpha_j - r_{j,k+1}}{\delta} + \sum_{i=1}^{L-1} \frac{r_{j,k+i} - r_{j,k+i+1}}{\delta_{i+1}} + r_{j,k+L+1} = \frac{\alpha_j}{\delta} + \sum_{i=1}^{L-1} \left(\frac{1}{\delta_{i+1}} - \frac{1}{\delta_i} \right) r_{j,k+i}. \quad (9)$$

This, together with a *dual fitting* argument similar to [15], imply the following.

Theorem 1 *Let $(\xi_f \geq 1, \xi_c)$ be such that ξ_c is an upper bound on the solution of the following maximization program for every k.*

$$\begin{aligned} \text{maximize} \quad & \frac{\sum_{j=1}^{k} \left(\frac{\alpha_j}{\delta} + \sum_{i=1}^{L-1} \left(\frac{1}{\delta_{i+1}} - \frac{1}{\delta_i} \right) r_{j,k+i} \right) - \xi_f f}{\sum_{i=1}^{k} d_i} \quad (10) \\ \text{subject to} \quad & (2), (3), (4), (7), (8), (6) \end{aligned}$$

Then, the MYZ$_L$ algorithm is a (ξ_f, ξ_c)-approximation algorithm for UFLP.

In the following theorem, we analyze the factor-revealing LP (10) and derive the main result of [20]. In order to do this, we need to set the values of δ_i's. Here, for simplicity of computations, we set δ_i to $\delta \cdot \frac{L-i}{L-1}$; however, it is easy to observe that any choice of δ_i's such that the limit of $\max_i(\delta_{i+1} - \delta_i)$ as L tends to infinity is zero, will also work. The proof is given in Appendix B.

Theorem 2 *Let (γ_f, γ_c) be a pair given by the maximization program (1) in Theorem A, and δ be an arbitrary number greater than or equal to 1. Then for every ϵ, if L is a sufficiently large constant, the MYZ$_L$ algorithm is a $(\gamma_f + \ln(\delta) + \epsilon, 1 + \frac{\gamma_c - 1}{\delta})$-approximation algorithm for UFLP.*

Furthermore, in the approximation ratio analysis we do not need L too big but just a constant to have $(L-1)(\delta^{1/(L-1)} - 1)$ close to $\ln(\delta)$ on first few digits, which still leads to a 1.52 approximation ratio. Thus, we have

Corollary 3 *The MYZ$_L$ algorithm, for a constant L, has a 1.52 approximation ratio and has a running time equal the JMS running time plus $O(m)$. In particular, by using the Thorup [28] implementation in its first phase, the MYZ$_L$ algorithm runs in an $O(m \ln(n))$ time.*

4 Linear-Cost Facility Location Problem

In the *linear-cost facility location problem*, we are given a set $\mathcal{F}$ of facilities, a set $\mathcal{D}$ of clients, metric connection costs c_{ij} between client j to facility i, and nonnegative values a_i and b_i for each facility that define the facility cost functions for opening facility i and connecting k clients to it as follows:

$$f_i(k) = \begin{cases} 0 & k = 0 \\ a_i k + b_i & k > 0 \end{cases}$$

We denote this linear-cost facility location problem as $FLP(a, b, c)$. Clearly, the regular UFLP is a special case of linear-cost FLP with $a_i = 0$, i.e., $FLP(0, b, c)$. Furthermore, it is straightforward to see that $FLP(a, b, c)$ is equivalent to the regular UFLP $FLP(0, b, a + c)$, that is, we can shift the variable facility cost into the connection cost. Thus, the linear-cost FLP can be solved using any algorithm for the UFLP, and the overall approximation ratio will be the same. However, for applications in the next section, we need trade-off approximation factors of the algorithm (as defined in Definition 1).

It is not necessarily true that applying a (γ_f, γ_c)-approximation algorithm for the UFLP on the instance $FLP(0, b, a + c)$ will give a (γ_f, γ_c)-approximate solution for $FLP(a, b, c)$. However, we will show that the JMS algorithm has this property. The following lemma, whose proof is given in Appendix C, generalizes Theorem A for the linear-cost FLP.

Lemma 4 *Let (γ_f, γ_c) be a pair obtained from the factor-revealing LP (1). Then applying the JMS algorithm on the instance $FLP(0, b, a+c)$ will give a (γ_f, γ_c)-approximate solution for $FLP(a, b, c)$.*

The above lemma and Theorem 9 in [15] give us the following corollary, which will be used in the next section.

Corollary 5 *There is a $(1, 2)$-approximation algorithm for the linear-cost FLP.*

It is worth noting that Theorem 2 can be easily generalized for the linear-cost FLP. The only trick is to scale up both a and b in the first phase by a factor of δ, and scale them both down in the second phase. The rest of the proof is almost the same as the proof of Lemma 4.

5 Reduction between Facility Location Problems

In the rest of this paper, we will show how the soft-capacitated facility location problem and other problems can be reduced to the linear-cost FLP or UFLP. First we define the concept of reduction between facility location problems.

Definition. A reduction from a facility location problem $\mathcal{A}$ to another facility location problem $\mathcal{B}$ is an efficient procedure R that maps every instance $\mathcal{I}$ of $\mathcal{A}$ to an instance $R(\mathcal{I})$ of $\mathcal{B}$. This procedure is called a (σ_f, σ_c)-reduction if the following conditions hold.

1. For any instance $\mathcal{I}$ of $\mathcal{A}$ and any feasible solution for $\mathcal{I}$ with facility cost $F^*_{\mathcal{A}}$ and connection cost $C^*_{\mathcal{A}}$, there is a corresponding solution for the instance $R(\mathcal{I})$ with facility cost $F^*_{\mathcal{B}} \leq \sigma_f F^*_{\mathcal{A}}$ and connection cost $C^*_{\mathcal{B}} \leq \sigma_c C^*_{\mathcal{A}}$.
2. For any feasible solution for the instance $R(\mathcal{I})$, there is a corresponding feasible solution for $\mathcal{I}$ whose total cost is at most as much as the total cost of the original solution for $R(\mathcal{I})$. In other words, the FLP instance $R(\mathcal{I})$ is an over-estimate of the FLP instance $\mathcal{I}$.

Theorem 6 *If we have a (σ_f, σ_c)-reduction from a facility location problem $\mathcal{A}$ to another facility location problem $\mathcal{B}$, and a (γ_f, γ_c)-approximation algorithm for $\mathcal{B}$, then there is a $(\gamma_f\sigma_f, \gamma_c\sigma_c)$-approximation algorithm for $\mathcal{A}$.*

Proof : On an instance $\mathcal{I}$ of the problem $\mathcal{A}$, we compute $R(\mathcal{I})$, run the (γ_f, γ_c)-approximation algorithm for $\mathcal{B}$ on $R(\mathcal{I})$, and output the corresponding solution for $\mathcal{I}$. In order to see why this is a $(\gamma_f \sigma_f, \gamma_c \sigma_c)$-approximation algorithm for $\mathcal{A}$, let SOL denote an arbitrary solution for $\mathcal{I}$, ALG denote the solution that the above algorithm finds, and $F^*_{\mathcal{P}}$ and $C^*_{\mathcal{P}}$ ($F^{ALG}_{\mathcal{P}}$ and $C^{ALG}_{\mathcal{P}}$, respectively) denote the facility and connection costs of SOL (ALG, respectively) when viewed as a solution for the problem $\mathcal{P}$ ($\mathcal{P} = \mathcal{A}, \mathcal{B}$). By the definition of (σ_f, σ_c)-reductions and (γ_f, γ_c)-approximation algorithms we have

$$F^{ALG}_{\mathcal{A}} + C^{ALG}_{\mathcal{A}} \leq F^{ALG}_{\mathcal{B}} + C^{ALG}_{\mathcal{B}} \leq \gamma_f F^*_{\mathcal{B}} + \gamma_c C^*_{\mathcal{B}} \leq \gamma_f \sigma_f F^*_{\mathcal{A}} + \gamma_c \sigma_c C^*_{\mathcal{A}},$$

which completes the proof of the lemma. $\square$

We will see examples of reductions in the rest of this paper.

5.1 Soft-capacitated FLP

Recall the definition of the soft-capacitated FLP. In this subsection, we give a 2-approximation algorithm for Soft-capacitated FLP by using a reduction from the soft-capacitated FLP to the linear-cost FLP.

Theorem 7 *There is a* 2*-approximation algorithm for the soft-capacitated FLP.*

Proof : We use the following reduction: Construct an instance of the linear-cost FLP, where we have the same sets of facilities and clients. The connection costs remain the same. However, the facility cost of the ith facility is $(1 + \frac{k-1}{u_i}) f_i$ if $k \geq 1$ and 0 if $k = 0$. Note that, for every $k \geq 1$, $\lceil \frac{k}{u_i} \rceil \leq 1 + \frac{k-1}{u_i} \leq 2 \cdot \lceil \frac{k}{u_i} \rceil$. Therefore, it is easy to see that this reduction is a $(2, 1)$-reduction. By Lemma 5, there is a $(1, 2)$-approximation algorithm for linear-cost FLP, which together with Theorem 6 completes the proof. $\square$

Furthermore, we now illustrate that a natural linear programming relaxation of the Soft-capacitated FLP has an integral gap of 2. In this sense, our analysis for the Soft-capacitated FLP is tight. It is known that the Soft-capacitated FLP can be formulated as the following integer program:

$$\begin{aligned}
\text{minimize} \quad & \sum_{i \in \mathcal{F}} f_i y_i + \sum_{i \in \mathcal{F}} \sum_{j \in \mathcal{C}} c_{ij} x_{ij} && (11)\\
\text{subject to} \quad & \forall\, i \in \mathcal{F}, j \in \mathcal{C}: \; x_{ij} \leq y_i \\
& \forall\, i \in \mathcal{F}: \; \sum_{j \in \mathcal{C}} x_{ij} \leq u_i y_i \\
& \forall\, j \in \mathcal{C}: \; \sum_{i \in \mathcal{F}} x_{ij} = 1 \\
& \forall\, i \in \mathcal{F}, j \in \mathcal{C}: \; x_{ij} \in \{0, 1\} && (12)\\
& \forall\, i \in \mathcal{F}: \; y_i \text{ is a nonnegative integer} && (13)
\end{aligned}$$

In a natural linear program relaxation, we replace constraint $x_{ij} \in \{0, 1\}$ of (12) by $x_{ij} \geq 0$, and remove constraint (13).

Here we see that even if we relax only constraint (13), the integral gap is already 2. Consider an instance of the SFLP, in which we are given only one potential facility i, and $k \geq 2$ clients need to be connected to it. Assume that the capacity of facility i is $u_i = k-1$, the facility cost $f_i = 1$, the connection cost $c_{ij} = 0$ for all j. It is clear that the optimal cost must be $2f_i$. However, after relaxing constraint (13), the optimal cost is $(1 + \frac{1}{k-1})f_i$. Therefore, the integral gap between the integer program and its relaxation is $\frac{2k}{k-1}$ which tends to 2 when k large enough.

5.2 A generalization of the Soft-capacitated FLP

In this subsection, we consider the following generalization of the soft-capacitated facility location problem. This problem is the same as soft-capacitated FLP except that if r copies of facility i are open, then the facility cost is $g(r)a_i$ where $g(r)$ is a concave function of $r \geq 0$. In other words, concave soft-capacitated FLP is a special case of GFLP in which the facility cost functions are of the form $f_i(x) = a_i g(\lceil x/u_i \rceil)$ for constants a_i, u_i and concave function g. It is also a special case of the so-called stair-case cost facility location problem [12]. On the other hand, it is a common generalization of the soft-capacitated FLP (when $g(r) = r$) and concave-cost FLP defined by [13] (when $u_i = 1$ for all i). The concave-cost FLP is a special case of GFLP in which facility cost functions are required to be concave. The main result of this subsection is the following:

Theorem 8 *The concave soft-capacitated FLP is* $(\frac{g(2)}{g(1)}, 1)$ *reducible to the linear-cost FLP.*

The proof of the above theorem is given in Appendix D. We first show that the concave soft-capacitated FLP is $(\frac{g(2)}{g(1)}, 1)$ reducible to the concave-cost FLP. Then we show that the latter is equivalent to the linear-cost FLP. By Theorem 8, a good approximation algorithm for linear-cost FLP would imply a good approximation for the concave soft-capacitated FLP. Furthermore, we can show that this reduction is tight as we proved for the soft-capacitated FLP.

6 Other Facility Location Problems

In this section, we present a couple more applications of Theorem 6.

6.1 Capacitated cable facility location (CCFLP)

CCFLP was introduced by Ravi and Sinha [22]. In this problem, each client can be served by a open facility either by directly connect to the facility, or connect to another client which is served by the facility. The connection is made by construct cables on the edges and each cable has a capacity u, the upper bound on the total demands routed by a cable. We wish to open facilities and construct a network of cables, such that every client is served by some open facility and all cable capacity are observed. This problem can be easily reduced to UFLP if $u = 1$ and each client has a demand of 1. If all the clients have uniform demands, Ravi and Sinha [22] show that CCFLP can be approximated by a factor of $\rho_{UFLP} + \rho_{ST}$ where ρ_{UFLP} and ρ_{ST} denote the approximation ratio for UFLP and the Steiner Tree problem. The current best ratios are $\rho_{UFLP} \leq 1.52$ and

$\rho_{ST} \leq 1.55$ [23]. When the demands are non-uniform, CCFLP can be approximated by a factor of $2\rho_{UFLP} + \rho_{ST} \leq 4.59$.

Corollary 9 *The capacitated cable facility location problem (CCFLP) can be approximated by a factor of* 3.97 *when the demands of clients are non-uniform.*

Proof : It has been proved in [22] that, if we have a (R_f, R_c)-approximation algorithm for UFLP, and a ρ_{ST}-approximation algorithm for the Steiner tree problem, then CCFLP can be approximated by a factor of $\max(R_f, 2R_c) + \rho_{ST}$ when the demands of clients are non-uniform.

It is known that $\rho_{ST} \leq 1.55$. If we solve UFLP using the MYZ algorithm, then by Theorem 6, we get that

$$\max(R_f, 2R_c) + \rho_{ST} \leq \max(\gamma_f + \ln\delta, 2(1 + \frac{\gamma_c - 1}{\delta}) + \rho_{ST} \leq 3.97$$

if we choose $(\gamma_f, \gamma_c) = (1.11, 1.78)$ and $\delta = 3.7$. □

6.2 Load-balanced facility location (LBFLP)

In LBFLP, each open facility i is required to serve at least L_i clients, and of course each client must be served by exactly one facility. This problem was introduced independently by Guha, Meyerson and Munagala [11] and by Karger and Minkoff [17]. They reduce LBFLP to UFLP. Assume that UFLP can be approximated by a factor of ρ. Both papers get a bicriteria $(\frac{1+a}{1-a}\rho, a)$-approximation for LBFLP, i.e, the algorithm produce a solution in which each facility i serve at least aL_i clients for some $a < 1$, and the total cost is at most $\frac{1+a}{1-a}\rho$ times the optimal cost. Their result can be used to obtain constant factor approximation algorithms for the connected facility location problem and some network design problems, in particular the access network design problem. This, together with $\rho = 1.52$, lead to a bicriteria approximation ratio for LBFLP equal $(4.56, 0.5)$ or $(3.04, 1/3)$.

In [11] and [17], it has been shown that any instance of LBFLP with total cost $C^* + F^*$ can be reduced to an instance of UFLP with total cost at most $(1+\lambda)(C^* + F^*)$, and any solution for the reduced UFLP can be translated to the original LBFLP while guarantee that each open facility serves at least aL_i clients, where $\lambda = \frac{2a}{1-a}$.

However, the same reduction shows that the reduced UFLP instance has an solution with total facility cost at most $F^* + \lambda C^*$ and total connection cost at most C^*. Therefore, we have the following

Corollary 10 *There is a bicriteria* $(\frac{2a}{1-a}\gamma_f + \gamma_c, a)$*-approximation for the load-balanced facility location problem (LBFLP), if UFLP can be approximated by a factor of* (γ_f, γ_c)*. In particular, when setting* $a = 1/2$ *and using* $(\gamma_f, \gamma_c) = (1, 2)$*, we have the approximation ratio equal* $(4, 0.5)$*; when setting* $a = 1/3$ *and using* $(\gamma_f, \gamma_c) = (1.11, 1.78)$*, we have the approximation ratio equal* $(2.89, 1/3)$*.*

Acknowledgement We would like to thank Asaf Levin for pointing out that our analysis of the 2-approximation algorithm for the soft-capacitated facility location problem is tight.

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] P. Bauer and R. Enders. A Capacitated Facility Location Problem with Integer Decision Variables. ISMP 1997.

[3] 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.

[4] F.A. Chudak. Improved approximation algorithms for uncapacitated 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.

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

[6] F.A. Chudak and D.B. Shmoys. Improved approximation algorithms for the capacitated facility location problem. In *Proc. 10th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 875-876, 1999.

[7] G. Cornuejols, G.L. Nemhauser, and L.A. Wolsey. The uncapacitated facility location problem. In P. Mirchandani and R. Francis, editors, *Discrete Location Theory*, pages 119–171. John Wiley and Sons Inc., 1990.

[8] R. Gandhi, S. Khuller, S. Parthasarathy and A. Srinivasan. Dependent Rounding on Bipartite Graphs *43rd Conference Foundations of Computer Science (FOCS)*, 2002.

[9] S. Guha, R. Hassin, S. Khuller and E. Or. Capacitated vertex covering with application. *SODA*, 2002.

[10] S. Guha and S. Khuller. Greedy strikes back: Improved facility location algorithms. *Journal of Algorithms*, 31:228–248, 1999.

[11] S. Guha, A. Meyerson, and K. Munagala. Hierarchical placement and network design problems. In *Proceedings of the 41th Annual IEEE Symposium on Foundations of Computer Science*, 2000.

[12] K. Holmberg. Solving the staircase cost facility location problem with decomposition and piecewise linearization. *European Journal of Operational Research*, 74:41-61, 1994.

[13] M. Hajiaghayi, M. Mahdian, and V.S. Mirrokni. The facility location problem with concave cost functions. Unpublished manuscript, 2002.

[14] K. Jain, M. Mahdian, Markakis E., A. Saberi, and V.V. Vazirani. Approximation algorithms for facility location via dual fitting with factor-revealing lp. Unpublished manuscript, 2002.

[15] K. Jain, M. Mahdian, and A. Saberi. A new greedy approach for facility location problems. In *STOC 2002*, 2002.

[16] K. Jain and V.V. Vazirani. Approximation algorithms for metric facility location and k-median problems using the primal-dual schema and lagrangian relaxation. *Journal of the ACM*, 48:274–296, 2001.

[17] D. Karger and M. Minkoff. Building steiner trees with incomplete global knowledge. In *Proceedings of the 41st Annual IEEE Symposium on Foundations of Computer Science*, 2000.

[18] 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.

[19] 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.

[20] M. Mahdian, Y. Ye, and J. Zhang. Improved approximation algorithms for metric facility location problems. In *5th International Workshop on Approximation Algorithms for Combinatorial Optimization (APPROX 2002)*, 2002.

[21] M. Pal, E. Tardos and T. Wexler. Facility location with hard capacities. *FOCS*, 2001.

[22] R. Ravi and A. Sinha. Integrated logistics: Approximation algorithms combining facility location and network design. In *Proceedings of the 9th Conference on Integer Programming and Combinatorial Optimization (IPCO 2002)*, 2002.

[23] G. Robins and A. Zelikovsky. Improved steiner tree approximation in graphs. In *Proceedings of the 10th ACM-SIAM symposium on Discrete Algorithms (SODA 1999)*, 1999.

[24] D.B. Shmoys. Approximation algorithms for facility location problems. In K. Jansen and S. Khuller, editors, *Approximation Algorithms for Combinatorial Optimization*, volume 1913 of *Lecture Notes in Computer Science*, pages 27–33. Springer, Berlin, 2000.

[25] 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.

[26] M. Sviridenko. An 1.582-approximation algorithm for the metric uncapacitated facility location problem. to appear in the Ninth Conference on Integer Programming and Combinatorial Optimization, 2002.

[27] M. Thorup. Quick k-median, k-center, and facility location for sparse grphas. In *Proceedings of the 28th ICALP*, LNCS 2076, pages 249-260, 2001.

[28] M. Thorup. Quick and good facility location. manuscript, AT&T Labs, 2002, to appear in *SODA 2003*.

A Performance of Throup's Algorithm

In [28], Throup shows that his algorithm, which runs in an $O(m\ln(n)/\epsilon)$ time, has almost the same performance guarantee with that of the JMS algorithm. Here we show that the it is also true in terms of the trade-off approximation factor, i.e., Throup's algorithm generates a close-to-(γ_f, γ_c)-approximation solution for UFLP, where (γ_f, γ_c) is a pair from LP (1). From [28], Throup's algorithm is a (γ'_f, γ'_c)-approximation algorithm for the UFLP if $\gamma'_f > 1$ be fixed and $\gamma'_c := \sup_k\{z_k\}$, where z_k is the solution of the following linear program:

$$
\begin{array}{ll}
\text{maximize} & \dfrac{\sum_{i=1}^k \alpha_i - \gamma'_f f}{\sum_{i=1}^k d_i} \\
\text{subject to} & \forall\, 1 \le i < k: \ \alpha_i \le \alpha_{i+1} \\
& \forall\, 1 \le j < i < k: \ r_{j,i} \ge r_{j,i+1} \\
& \forall\, 1 \le j < i \le k: \ \alpha_i \le (1+\epsilon) r_{j,i} + d_i + d_j \\
& \forall\, 1 \le i \le k: \ \displaystyle\sum_{j=1}^{i-1} \max(r_{j,i} - d_j, 0) + \sum_{j=i}^{k} \max(\alpha_i - d_j, 0) \le f \\
& \forall\, 1 \le j \le i \le k: \ \alpha_j, d_j, f, r_{j,i} \ge 0
\end{array}
\tag{14}
$$

where $\epsilon > 0$ is a small constant. It is quite clear that for any given $\gamma_f = \gamma'_f/(1+\epsilon) \ge 1$, the objective value of LP (1) γ_c is bounded below by $\gamma' c/(1+\epsilon)$. Therefore, if JMS algorithm is a (γ_f, γ_c)-approximation algorithm, then Throup's algorithm is a $(\gamma_f(1+\epsilon), \gamma_c(1+\epsilon))$-approximation algorithm for UFLP. For choosing ϵ sufficiently small but a constant, the (trade-off) performance guarantees of JMS algorithm and Throup's algorithm are almost the same.

B Proof of Theorem 2

Proof : Since inequalities of the factor-revealing LP (10) are a superset of the inequalities of the factor-revealing LP (1), therefore by Theorem A and the definition of (γ_f, γ_c), we have

$$
\sum_{j=1}^{k} \alpha_j \le \gamma_f \delta f + \gamma_c \sum_{j=1}^{k} d_j \tag{15}
$$

By the fifth inequality of the factor-revealing LP (10), we have that for all $i = 1, \ldots, L$,

$$
\sum_{j=1}^{k} r_{j,k+i} \le \sum_{j=1}^{k} \max(r_{j,k+i} - d_j, 0) + \sum_{j=1}^{k} d_j \le \delta_i f + \sum_{j=1}^{k} d_j.
$$

Therefore,

$$
\sum_{j=1}^{k} \left(\frac{\alpha_j}{\delta} + \sum_{i=1}^{L-1} \left(\frac{1}{\delta_{i+1}} - \frac{1}{\delta_i} \right) r_{j,k+i} \right)
$$

$$
\begin{aligned}
&= \frac{1}{\delta}(\sum_{j=1}^{k}\alpha_j) + \sum_{i=1}^{L-1}\left((\frac{1}{\delta_{i+1}} - \frac{1}{\delta_i})\sum_{j=1}^{k} r_{j,k+i} \right) \\
&\leq \frac{1}{\delta}(\gamma_f \delta f + \gamma_c \sum_{j=1}^{k} d_j) + \sum_{i=1}^{L-1}\left((\frac{1}{\delta_{i+1}} - \frac{1}{\delta_i})(\delta_i f + \sum_{j=1}^{k} d_j) \right) \\
&= \gamma_f f + \frac{\gamma_c}{\delta}\sum_{j=1}^{k} d_j + \sum_{i=1}^{L-1}(\frac{\delta_i}{\delta_{i+1}} - 1) f + (\frac{1}{\delta_L} - \frac{1}{\delta_1})\sum_{j=1}^{k} d_j \\
&= \left(\gamma_f + (L-1)(\delta^{1/(L-1)} - 1)\right) f + (\frac{\gamma_c}{\delta} + 1 - \frac{1}{\delta})\sum_{j=1}^{k} d_j.
\end{aligned}
$$

This, together with Theorem 1 shows that the MYZ$_L$ is a $(\gamma_f + (L-1)(\delta^{1/(L-1)} - 1), 1 + \frac{\gamma_c - 1}{\delta})$-approximation algorithm for UFLP. The fact that the limit of $(L-1)(\delta^{1/(L-1)} - 1)$ as L tends to infinity is $\ln(\delta)$ completes the proof. $\square$

C Proof of Lemma 4

Proof : Let SOL be an arbitrary solution for $FLP(a, b, c)$, which can also be viewed as a solution for $FLP(0, b, \bar{c})$ for $\bar{c} = c + a$. Consider a facility f that is open in SOL, and the set of clients connected to it in SOL. Let k denote the number of these clients, $f(k) = ak + b$ (for $k > 0$) be the facility cost function of f, and $\bar{d}_j$ denote the connection cost between client j and the facility f in the modified instance $FLP(0, b, a + c)$. Therefore, $d_j = \bar{d}_j - a$ is the corresponding connection cost in the original instance ($FLP(a, b, c)$). Recall [15] the definition of α_j and r_{ij} in the factor-revealing LP. It is proved [15] that $\alpha_i \leq r_{j,i} + \bar{d}_j + \bar{d}_i$. We strengthen this inequality as follows.

Claim 11 $\alpha_i \leq r_{j,i} + d_j + d_i$

It is true if $\alpha_i = \alpha_j$ since it happens only if $r_{j,i} = \alpha_j$. Otherwise, consider clients i and $j (< i)$ at time $t = \alpha_i - \epsilon$. Let s be the facility j is assigned to at time t. By triangle inequality, we have

$$\bar{c}_{si} = c_{si} + a_s \leq c_{sj} + d_i + d_j + a_s = \bar{c}_{sj} + d_i + d_j \leq r_{j,i} + d_i + d_j.$$

On the other hand $\alpha_i \leq \bar{c}_{si}$ since otherwise i could have connected to facility s at a time earlier than t.

It is also known [15] that

$$\sum_{j=1}^{i-1} \max(r_{j,i} - \bar{d}_j, 0) + \sum_{j=i}^{k} \max(\alpha_i - \bar{d}_j, 0) \leq b.$$

Notice that $\max(a - x, 0) \geq \max(a, 0) - x$ if $x \geq 0$. Therefore, we have

$$\sum_{j=1}^{i-1} \max(r_{j,i} - d_j, 0) + \sum_{j=i}^{k} \max(\alpha_i - d_j, 0) \leq b + ka. \tag{16}$$

Claim 11 and Inequality 16 show that the values α_j, r_{ij}, d_j, a, and b constitute a feasible solution of the following optimization program.

$$
\begin{aligned}
\text{maximize} \quad & \frac{\sum_{i=1}^{k} \alpha_i - \gamma_f(ak+b)}{\sum_{i=1}^{k} d_i} && (17) \\
\text{subject to} \quad & \forall\, 1 \le i < k: \ \alpha_i \le \alpha_{i+1} \\
& \forall\, 1 \le j < i < k: \ r_{j,i} \ge r_{j,i+1} \\
& \forall\, 1 \le j < i \le k: \ \alpha_i \le r_{j,i} + d_i + d_j \\
& \forall\, 1 \le i \le k: \ \sum_{j=1}^{i-1} \max(r_{j,i} - d_j, 0) + \sum_{j=i}^{k} \max(\alpha_i - d_j, 0) \le b + ka \\
& \forall\, 1 \le j \le i \le k: \ \alpha_j, d_j, a, b, r_{j,i} \ge 0
\end{aligned}
$$

However, it is clear that the above optimization program and the factor-revealing LP 1 are equivalent. This completes the proof of this lemma. $\square$

D Proof of Theorem 8

The theorem is established by the following lemmas which show the reductions between the concave soft-capacitated FLP, the concave-cost FLP and the linear-cost FLP.

Lemma 12 *The concave soft-capacitated FLP is* $(\frac{g(2)}{g(1)}, 1)$ *reducible to the concave-cost FLP.*

Proof : Given an instance of the concave soft-capacitated FLP, recall that the facility cost of i is $g(\lceil \frac{k}{u_i} \rceil) a_i$. We use the following reduction: Construct an instance of the linear-cost FLP, where we have the same sets of facilities and clients. The connection costs remain the same. However, the facility cost of the ith facility is $\left(g(r) + (g(r+1) - g(r))(r-1) + \frac{k-1}{u_i}(g(r+1) - g(r))\right) a_i$ if $(r-1)u_i + 1 \le k \le ru_i$ and 0 if $k = 0$. By the concavity of the function $g(r)$, we must have

$$\frac{g(r+1)}{g(r)} \le \frac{g(2)}{g(1)} \qquad \text{for} \quad r \ge 1.$$

Therefore, when $(r-1)u_i + 1 \le k \le ru_i$, we get

$$\left(g(r) + (g(r+1) - g(r))(r-1) + \frac{k-1}{u_i}(g(r+1) - g(r))\right) a_i \le g(r+1)a_i \le \frac{g(2)}{g(1)} g(r) a_i.$$

On the other hand, since $g(r+1) \ge g(r)$, we have

$$\left(g(r) + (g(r+1) - g(r))(r-1) + \frac{k-1}{u_i}(g(r+1) - g(r))\right) a_i \ge g(r) a_i,$$

This completes the proof of the lemma. $\square$

Moreover, we will show a simple $(1,1)$-reduction from the concave-cost FLP to the linear-cost FLP. This, together with the above lemma, reduce the concave soft-capacitated facility location problem to the linear-cost FLP.

Lemma 13 *The concave-cost FLP is equivalent to linear-cost FLP.*

Proof : We use the following reduction: Corresponding to a facility with facility cost function $f(k)$, we put n copies of this facility (where n is the number of cities), and let the facility cost function of the i'th copy be

$$f^{(i)}(k) = \begin{cases} i \cdot f(i-1) - (i-1) \cdot f(i) + (f(i) - f(i-1)) \cdot k & \text{if } k > 0 \\ 0 & \text{if } k = 0 \end{cases}$$

In other words, the facility cost function is the line that passes through the points $(i-1, f(i-1))$ and $(i, f(i))$. The set of clients, and the connection costs between clients and facilities are unchanged. We prove that this reduction is a $(1,1)$-reduction. Let's denote the concave-cost FLP instance by $\mathcal{I}$, and the linear-cost FLP instance constructed above by $R(\mathcal{I})$.

For any feasible solution SOL for $\mathcal{I}$, we can construct a feasible solution for $R(\mathcal{I})$ as follows: If a facility is open and k clients are connected to it in SOL, we open the k'th copy of the corresponding facility in $R(\mathcal{I})$, and connect the clients to it. Since the cost of a facility in $\mathcal{I}$ when k clients are connected to it is equal to the cost of the k'th copy of the corresponding facility in $R(\mathcal{I})$ when k clients are connected to it, therefore the facility and connection costs of the solution constructed for $R(\mathcal{I})$ is the same as those of SOL.

On the other hand, consider an arbitrary feasible solution SOL for the instance $R(\mathcal{I})$. We construct a solution for $\mathcal{I}$ as follows. For any facility f, if at least one of the copies of f is open in SOL, we open f and connect all clients that were served by a copy of f in SOL to it. We show that this does not increase the total cost of the solution: Assume the i_1'th, i_2'th, $\ldots$, and i_s'th copies of f were open in SOL, serving $k_1, k_2, \ldots, k_s$ clients, respectively. By concavity of f, and the fact $f^{(i)}(k) \geq f^{(k)}(k) = f(k)$ for every i, we have

$$f(k_1 + \cdots + k_s) \leq f(k_1) + \cdots + f(k_s) \leq f^{(i_1)}(k_1) + \cdots + f^{(i_s)}(k_s).$$

This shows that the facility cost of the solution we constructed for $\mathcal{I}$ is at most the facility cost of SOL. $\square$