Improved Approximation Algorithm for Capacitated Facility Location with Uniform Facility Cost

Mong-Jen Kao
Department of Computer Science, National Yang-Ming Chiao-Tung University, Hsinchu, Taiwan

Abstract

We consider the hard-capacitated facility location problem with uniform facility cost (CFL-UFC). This problem arises as an indicator variation between the general CFL problem and the uncapacitated facility location (UFL) problem, and is related to the profound capacitated k-median problem (CKM).

In this work, we present a rounding-based 4-approximation algorithm for this problem, built on a two-staged rounding scheme that incorporates a set of novel ideas and also techniques developed in the past for both facility location and capacitated covering problems. Our result improves the decades-old LP-based ratio of 5 for this problem due to Levi et al. since 2004. We believe that the techniques developed in this work are of independent interests and may further lead to insights and implications for related problems.

2012 ACM Subject Classification Theory of computation → Facility location and clustering

Keywords and phrases Capacitated facility location, Hard capacities, Uniform facility cost

Digital Object Identifier 10.4230/LIPIcs.ISAAC.2023.45

Related Version *Full Version*: `https://arxiv.org/abs/2102.06613`

Funding *Mong-Jen Kao*: Supported in part by National Science and Technology Council (NSTC), Taiwan, under Grants 111-2221-E-A49-118-MY3, 112-2628-E-A49-017-MY3, and 112-2634-F-A49-001-MBK.

1 Introduction

We consider the facility location problem with hard capacities and uniform facility cost (CFL-UFC). In this problem, we are given a set $\mathcal{F}$ of facilities, a set $\mathcal{D}$ of clients, and a distance metric c defined over $\mathcal{F} \cup \mathcal{D}$. Each facility $i \in \mathcal{F}$ is associated with a uniform open cost w and a capacity u_i, which is the number of clients facility i can serve when opened up. The cost of assigning a client to a facility is equal to the distance between them. The goal of this problem is to compute a set of facilities $A \subseteq \mathcal{F}$ to open up and an assignment function $h \colon \mathcal{D} \to A$ that respects the capacity limits of the facilities in A such that, the total facility open cost plus the assignment cost, $w \cdot |A| + \sum_{j \in \mathcal{D}} c_{j,h(j)}$, is minimized.

The CFL-UFC problem originates as an important variation of the classic capacitated facility location problem (CFL), in which the open cost of each facility can be non-uniform, and is deeply related to the profound capacitated k-median problem (CKM). To better illustrate the literature of CFL-UFC, in the following we start with the introduction for the CFL problem. Then we describe the implicit connection of CFL-UFC to CKM.

The classic problem of CFL was first addressed by Shmoys, Tardos, and Aardal [15]. Since then, almost all results for this problem were based on local search heuristics, e.g., [3,13,14,18], and the best ratio known for this problem is 5, due to Bansal et al. [3].

In contrast to the rich LP-based toolsets developed for the uncapacitated facility location problem (UFL), the fact that no LP-based algorithm with constant approximation guarantee for CFL was known for a long time was intriguing. In fact, devising an LP-based approximation with $O(1)$ guarantee for CFL was listed as one of ten open problems in the textbook due to Williamson and Shmoys [17]. This problem was resolved by the notable work of An et al. [2], in which a novel multi-commodity flow network (MFN) relaxation was presented. In a follow-up work, Kao [10] presented an iterative rounding approach and showed that, the integrality gap of the MFN relaxation is at most $(10+\sqrt{67})/2 \approx 9.0927$.

In the pursuit of settling down the approximability of the CFL problem, an important variation between the general CFL problem and the UFL problem is when we have uniform facility cost, i.e., CFL-UFC. This was studied by Levi et al. [11], in which a 5-approximation via LP-rounding was presented. On the other hand, Aardal et al. [1] presented a 4.562-approximation based on local search technique. Interestingly, the ratio of 5 due to Levi et al. [11] also remains the best LP-based guarantee for CFL-UFC for a surprisingly long period of time since 2004.

Comparing to local search algorithms, which often has the characteristics of being simple and elegant, LP-based methods have the irreplaceable advantage of being applicable to related problems to yield new results. One of such examples is the celebrating successful story of LMP-algorithms for UFL to be combined with bi-point rounding algorithms to yield state-of-the-art results for the uncapacitated k-median problem, see, e.g., [4–8, 12, 16].

So far, the phenomenon is very different for the capacitated version of facility location and k-median problems. One primary reason is that, very little is known regarding LP-based results for both CFL and CFL-UFC.

1.1 Our Contribution

In this work, we present an LP-based 4-approximation algorithm for CFL-UFC. Our main result is the following theorem.

▶ **Theorem 1.** *There is a rounding-based algorithm for CFL-UFC that produces a* 4*-approximation in polynomial-time.*

Theorem 1 improves the decades-old LP-based guarantee of 5 due to Levi et al. [11] since 2004 and shows that the integrality gap of natural LP for this problem is at most 4. The algorithm we present is built on a two-staged rounding scheme that incorporates a set of novel ideas together with results and techniques developed in the past for both facility location and capacitated covering problems [9, 11].

In the following, we describe the ideas we use to obtain the 4-approximation guarantee. We believe that, the techniques developed in this work are of independent interests and may lead to further insights and progress for further related problems.

Overview of our Algorithm and Techniques

The core part of our result can be seen as a delicate orchestration of rounding procedures for the large and small instances incurred in the LP solution. In our procedure, we aim at fractionally serving the clients while making sure that the rounded facilities of interests are sufficiently sparsely-loaded so that a reasonable round-up of the assignments can be made to fully-serve the clients. Intuitively, this is possible for small facilities as they are sparsely-loaded by default. The large facilities, however, do not allow such a round-up in general since they can be tightly-loaded by assignments made in the LP solution.

To overcome this issue, we introduce the concept of *client redistribution*: When the residue demand of a client drops below a target threshold, we discard the client and redistribute part of it to the large facilities in the vicinity, defined by the LP solution, to form the so-called "*outlier clients*." The outlier clients participate in the rounding process after created and act as normal clients except for that, there is no threshold for them to be discarded, and we guarantee that they will be fully-assigned for the final feasibility. Moreover, the way the outlier clients are created also guarantees that, the resulting assignment cost does not increase too much.

The concept of client redistribution resolves the assignment of the clients. However, when an outlier client is selected to form a cluster, we are no longer able to guarantee the overall rounding cost of the facilities, since the total facility value in that cluster can be arbitrarily small, rendering the rounding error unbounded. To prevent this undesirable situation, we introduce a matching-yielding LP technique and leave the rounding decisions for the *outlier clusters* as a global optimization problem to be resolved in the second stage of the algorithm.

In the second stage of the algorithm, we formulate the rounding problem of the remaining outlier clusters as a carefully designed assignment LP. We deploy a technique, that was originally developed for the capacitated covering problems [9], to show that, basic feasible solutions of this simple LP corresponds naturally to a matching from the non-integral facilities to the large facilities at which the outlier clients reside, and hence the rounding cost of these facilities can be bounded. Together this yields a bound for our final unconditional rounding.

Organization of this paper

This paper is organized as follows. In Section 2, we formally define CFL-UFC and describe preliminaries necessary to present our approximation algorithms. We present our approximation algorithm for CFL-UFC in Section 3 and the analysis in Section 4. Due to space limit, technical details and proofs omitted from the main content will be provided in the full version of this paper.

2 Preliminaries

In the CFL-UFC problem, we are given a set $\mathcal{F}$ of facilities, a set $\mathcal{D}$ of clients, and a distance metric c defined over $\mathcal{F} \cup \mathcal{D}$. Each $i \in \mathcal{F}$ is associated with a uniform open cost w and a capacity u_i, which is the number of clients facility i can serve when opened up. A feasible solution for CFL-UFC consists of a multiplicity function $y \colon \mathcal{F} \to \{0,1\}$ and an assignment function $x \colon \mathcal{F} \times \mathcal{D} \to \{0,1\}$ such that the following conditions are met:

- $\sum_{i\in\mathcal{F}} x_{i,j} \geq 1$, for any $j \in \mathcal{D}$, i.e., each client is assigned to some facility.
- $\sum_{j\in\mathcal{D}} x_{i,j} \leq u_i \cdot y_i$, for any $i \in \mathcal{F}$, i.e., the capacity limit of any facility is not violated.
- $x_{i,j} \leq y_i$, for any $i \in \mathcal{F}, j \in \mathcal{D}$, i.e., assignments can only be made to opened facilities.

The cost of the solution (x, y) is defined to be $\psi(x, y) := \sum_{i\in\mathcal{F}} w \cdot y_i + \sum_{i\in\mathcal{F},\, j\in\mathcal{D}} c_{i,j} \cdot x_{i,j}$. Note that, by properly rescaling the distance metric c, we may assume that $w = 1$. Given an instance $\Psi = (\mathcal{F}, \mathcal{D}, \mathbf{c}, \mathbf{u})$ of CFL-UFC, the goal of this problem is to compute a feasible solution (x, y) such that $\psi(x, y)$ is minimized.

LP relaxation and the definition of Vicinity

A natural LP relaxation for CFL-UFC and its dual LP is given below in Figure 1. It follows that, for optimal solutions (x, y) and $(\alpha, \beta, \Gamma, \eta)$ for LP-(N) and LP-(DN), $\alpha_j \geq c_{i,j}$ holds for any $i \in \mathcal{F}, j \in \mathcal{D}$ with $x_{i,j} > 0$. We will use the fact that α_j is a valid estimation on the assignment radius for any $j \in \mathcal{D}$ in x.

For the ease of presentation, in our algorithm, we use the following notion of vicinity that is defined with respect to any given assignment function x. For any $A \subseteq \mathcal{F}$ and any $j \in \mathcal{D}$, we use $N_{(A,x)}(j) := \left\{\, i \in A :\ x_{i,j} > 0 \,\right\}$ to denote the set of facilities in A to which j is assigned to in x. Similarly, for any $B \subseteq \mathcal{D}$ and any $i \in \mathcal{F}$, we use $N_{(B,x)}(i) := \left\{\, j \in B :\ x_{i,j} > 0 \,\right\}$ to denote the set of clients in B that is assigned to i in x.

LP-(N)

$$\min \sum_{i \in \mathcal{F}} y_i + \sum_{i \in \mathcal{F}, j \in \mathcal{D}} c_{i,j} \cdot x_{i,j}$$

$$\sum_{i \in \mathcal{F}} x_{i,j} \geq 1, \qquad \forall j \in \mathcal{D},$$

$$\sum_{j \in \mathcal{D}} x_{i,j} \leq u_i \cdot y_i, \qquad \forall i \in \mathcal{F},$$

$$0 \leq x_{i,j} \leq y_i, \qquad \forall i \in \mathcal{F}, j \in \mathcal{D},$$

$$0 \leq y_i \leq 1, \qquad \forall i \in \mathcal{F}.$$

LP-(DN)

$$\max \sum_{j \in \mathcal{D}} \alpha_j - \sum_{i \in \mathcal{F}} \eta_i$$

$$\alpha_j \leq \beta_i + \Gamma_{i,j} + c_{i,j}, \qquad \forall i \in \mathcal{F}, j \in \mathcal{D},$$

$$u_i \cdot \beta_i + \sum_{j \in \mathcal{D}} \Gamma_{i,j} \leq 1 + \eta_i, \qquad \forall i \in \mathcal{F},$$

$$\alpha_j, \beta_i, \Gamma_{i,j}, \eta_i \geq 0, \qquad \forall i \in \mathcal{F}, j \in \mathcal{D}.$$

Figure 1 LP relaxations for CFL-UFC.

3 4-Approximation for CFL-UFC

In the following, we describe our approximation algorithm $\mathcal{A}$ for CFL-UFC and prove Theorem 1. Let $\Psi = (\mathcal{F}, \mathcal{D}, \mathbf{c}, \mathbf{u})$ be an instance of CFL-UFC, and let $(\boldsymbol{x}', \boldsymbol{y}')$, $(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\Gamma}, \boldsymbol{\eta})$ be optimal solutions for LP-(N) and its dual LP-(DN) on Ψ.

Let $I := \left\{\, i \in \mathcal{F} : \ 0 < y'_i < \frac{1}{2} \,\right\}$ and $U := \left\{\, i \in \mathcal{F} : \ y'_i \geq \frac{1}{2} \,\right\}$ be the sets of small and large facilities. Let $J^{(I)}$ and $J^{(\leftrightarrow)}$ be the clients that are served merely by I and the clients that are served jointly by I and U, respectively. We round up the facilities in U directly and keep the assignments made to them unchanged. What remains is the rounding problem for I and $J^{(I)} \cup J^{(\leftrightarrow)}$.

Our rounding process for I and $J^{(I)} \cup J^{(\leftrightarrow)}$ consists of two stages. In the first stage, it proceeds in iterations to select clients and form clusters. Depending on the status of the client selected, the rounding decision for the cluster may be postponed. In the second stage, our rounding process formulates the rounding decisions of the postponed clusters as a global optimization problem and makes an overall rounding decision. In the following, we describe the two stages in details.

3.1 The First Stage of the Rounding Process

In this stage, the algorithm proceeds in iterations to form clusters. Let F' and D' be the set of facilities and the set of clients remained to be processed. Initially, $F' := I$ and $D' := J^{(I)} \cup J^{(\leftrightarrow)}$. The algorithm will maintain a rounded assignment function x^* during this stage. Initially $x^* := 0$.

In each iteration, the algorithm first checks if $\sum_{i \in F'} x'_{i,j} \geq 1/2$ holds for all $j \in D'$. Intuitively, from the LP constraints, this condition guarantees that

$$\sum_{i \in N_{(F', x')}(j)} y'_i \geq \sum_{i \in F'} x'_{i,j} \geq \frac{1}{2},$$

and there will be a decent amount of facility values to be aggregated in the vicinity of client j in F' for any $j \in D'$. If not, the algorithm makes it so by repeating the following steps:

1. Pick an arbitrary $j \in D'$ with $\sum_{i \in F'} x'_{i,j} < 1/2$.
2. Apply the procedure CREATE_OUTLIER(j),
   which we later describe, to create a set of *outlier clients* for j.
3. Remove the client j from D'.

Intuitively, each client j that is picked here will belong to the set $J^{(\leftrightarrow)}$, and via replacing j with its outlier copies created by the procedure CREATE_OUTLIER(j), the rounding cost for the remaining part of j will be charged to the large facilities to which j is assigned to, i.e., $N_{(U,x')}(j)$.

The algorithm additionally maintains two sets H and H', where H denotes the set of outlier clients created in this step and $H' \subseteq H$ denotes those that have been created but not yet processed by the rounding process. Initially $H := \emptyset$ and $H' := \emptyset$.

When the condition $\sum_{i \in F'} x'_{i,j} \geq 1/2$ holds for all $j \in D'$, the algorithm applies another procedure FORM_CLUSTER, which we will later describe, to select a client from $D' \cup H'$ and form a cluster centered at that client. Depending on whether or not the selected client is outlier, the rounding decision for the cluster created may be postponed to the second stage. The procedure then removes the corresponding parts of the cluster from the residual instance $(F', H', \boldsymbol{x}', \boldsymbol{y}')$.

When the procedure FORM_CLUSTER is done, for each client $j \in D' \cap J^{(I)}$ with $\sum_{i \in F'} x'_{i,j} < 1/2$, the algorithm removes j from D' and sets $x'_{i,j}$ to be zero for all $i \in F'$. Intuitively, for each client j that is picked in this step, the algorithm guarantees that, more than half of its demand has already been assigned to a rounded facility. Hence the remaining part can be discarded.

Then the algorithm iterates to the next iteration until $D' \cup H'$ becomes empty. The following high-level pseudo-code summarizes the first stage of our rounding process.

- Repeat until $D' \cup H' = \emptyset$, do
  1. Repeat until $\sum_{i \in F'} x'_{i,j} \geq 1/2$ for all $j \in D'$, do
     - Pick an arbitrary $j \in D'$ with $\sum_{i \in F'} x'_{i,j} < 1/2$.
     - Apply CREATE_OUTLIER(j) and remove j from D'.
  2. Apply the procedure FORM_CLUSTER().
  3. For all $j \in D' \cap J^{(I)}$ with $\sum_{i \in F'} x'_{i,j} < 1/2$,
     remove j from D' and set $x'_{i,j}$ to be zero for all $i \in F'$.

Note that, the algorithm guarantees the invariant that, the client picked in step 1 must belong to the set $D' \cap J^{(\leftrightarrow)}$. In the following we describe the two procedures CREATE_OUTLIER and FORM_CLUSTER in details.

The procedure CREATE_OUTLIER(j).

When this procedure is called, it relocates part of the remaining demand of j to facilities in $N_{(U,x')}(j)$, i.e., the large facilities in U for which j is assigned to in $\boldsymbol{x}'$, to form outlier clients in a way as if the demand were originated from these facilities. Then it updates the assignment function $\boldsymbol{x}'$ and the dual variables $\boldsymbol{\alpha}$ accordingly.

Before describing the detail of this procedure, we describe the intuitions in the following. The outlier clients created by this procedure will be used to replace the remaining part of j, and the construction will serve for two purposes.

- First, since $\sum_{i \in F'} x'_{i,j} < 1/2$ when this procedure is called, $\sum_{i \in N_{(F',x')}(j)} y'_i$ can be less than $1/2$. Therefore, when the outlier clients of j are selected to form clusters in later iterations of the rounding algorithm, we will use the facility values in $N_{(U,x')}(j)$ to amend the short deficits of $\sum_{i \in N_{(F',x')}(j)} y'_i$ compared to $1/2$.
- Second, via the construction scheme of outlier clients, we will charge part of the assignment costs of the outlier clients to the assignment costs of j to the large facilities in $N_{(U,x')}(j)$. We note that, this is reflected in the setting of the dual values of the outlier clients.

In the following, we describe the procedure in details.

Let $r'_j := \min\{ \sum_{i \in F'} x'_{i,j} \;,\; \sum_{i \in U} x'_{i,j} \}$ be the amount of residue demand of j to be redistributed. For each $w \in N_{(U,x')}(j)$, we create a client j_w at the facility w with demand

$$d_{j_w} := \frac{r'_j}{\sum_{i \in U} x'_{i,j}} \cdot x'_{w,j} \;\text{ and set }\; x'_{i,j_w} := \frac{d_{j_w}}{\sum_{k \in F'} x'_{k,j}} \cdot x'_{i,j}$$

for each $i \in N_{(F',x')}(j)$. We add j_w to both H and H' and set $\alpha_{j_w} := \alpha_j + c_{w,j}$ to reflect the relocation of j_w from j to w. Note that, this ensures that α_{j_w} is still a valid estimation on the assignment radius of j_w.

Note that, by construction, we have

$$\sum_{w \in N_{(U,x')}(j)} d_{j_w} = r'_j \quad \text{and} \quad \sum_{k \in N_{(F',x')}(j)} x'_{k,j_w} = d_{j_w},$$

i.e., the designated residue demand r'_j of j is fully redistributed as outlier clients and each j_w is fully-assigned to facilities in F'.

After the outlier client j_w is created for each $w \in N_{(U,x')}(j)$, the procedure removes j from D' and set $x'_{i,j}$ to be zero for all $i \in F'$. It is clear that, the above updates on $\boldsymbol{x}'$ does not violate the capacity constraints of the facilities in F'.

<u>The procedure FORM_CLUSTER().</u>

When this procedure is called, it selects a client $j \in D' \cup H'$ with the minimum α_j to form a cluster. Depending on the set to which j belongs, the procedure proceeds differently.

- If $j \in H'$, then a cluster centered at j with satellite facilities in $N_{(F',x')}(j)$ is formed. We use $B(j) := N_{(F',x')}(j)$ to denote the set of satellite facilities at this moment. The procedure removes j from H' and $B(j)$ from F'. The rounding decision for this cluster is postponed to the second stage of the algorithm.
- If $j \in D'$, the procedure selects a facility $i \in N_{(F',x')}(j)$ with the maximum capacity u_i. Since

  $$\sum_{k \in N_{(F',x')}(j)} y'_k \;\geq\; \sum_{k \in F'} x'_{k,j} \;\geq\; \frac{1}{2}$$

  holds when this procedure is called, we will fractionally round y'_i to 1/2 by aggregating both the facility values and the assignments from facilities in $N_{(F',x')}(j)$ to the selected facility i. This is done as follows.
  Let

  $$\delta_i := \left(\frac{1}{2} - y'_i \right) \cdot \frac{1}{\sum_{k \in N_{(F',x')}(j) \setminus \{i\}} y'_k}$$

  be the factor to relocate for each facility in $N_{(F',x')}(j) \setminus \{i\}$. For each facility $\ell \in N_{(F',x')}(j) \setminus \{i\}$, scale down y'_ℓ by $(1 - \delta_i)$. For each $k \in N_{(D' \cup H',x')}(\ell)$, further scale down $x'_{\ell,k}$ by $(1 - \delta_i)$ and increase $x^*_{i,k}$ by the same amount $x'_{\ell,k}$ is decreased in this step. Then the procedure increases $x^*_{i,k}$ by $x'_{i,k}$ for each $k \in D'$ and then removes i from F'.
  Intuitively, for each facility $\ell \in N_{(F',x')}(j) \setminus \{i\}$, we move δ_i fraction of its facility value and assignments to the facility i, and the assignment function $x^*_{i,k}$ records the rounded assignment of any $k \in D'$ to the selected facility i.

Properties Guaranteed in the 1^{st}-stage

To better illustrate how our rounding algorithm works, we summarize in the following the important properties guaranteed by the rounding procedure in the first stage. To be precise with notations, in the following, let $\boldsymbol{x}'^{(0)}$ denote the initial assignment, i.e., the initial $\boldsymbol{x}'$. For each outlier client $j \in H$, we extend the definition of $\boldsymbol{x}'^{(0)}$ to be the initial assignment of j when it was created. Let $(\boldsymbol{x}'^{(\text{II})}, \boldsymbol{y}'^{(\text{II})})$ denote the pair $(\boldsymbol{x}', \boldsymbol{y}')$ the algorithm has when it is about to enter the second stage, i.e., the pair $(\boldsymbol{x}', \boldsymbol{y}')$ when the first stage ends.

Let $G := \bigcup_{j \in H} B(j)$ be the set of satellite facilities whose rounding decisions are to be postponed in the second stage and $F^*_{\mathcal{D}'}$ denote the set of facilities that are selected and rounded up to $1/2$ by the procedure FORM_CLUSTER(). We have the following lemma.

▶ **Lemma 2.** *When first stage of the rounding algorithm ends, the following holds.*

- *For any $i \in G$, $\sum_{j \in \mathcal{D} \cup H} x'^{(II)}_{i,j} \le u_i \cdot y'^{(II)}_i$.*
- *For any $j \in J^{(I)}$, $\sum_{i \in I} x^*_{i,j} + \sum_{i \in G} x'^{(II)}_{i,j} > 1/2$.*
- *For any $j \in J^{(\leftrightarrow)}$, $\sum_{i \in I} x^*_{i,j} + \sum_{i \in G} x'^{(II)}_{i,j} + \sum_{i \in U} x'^{(0)}_{i,j} > 1/2$.*
- *For any outlier $j \in H$, $\sum_{i \in F^*_{\mathcal{D}'}} x^*_{i,j} + \sum_{i \in G} x'^{(II)}_{i,j} = \sum_{i \in I} x'^{(0)}_{i,j}$.*

Intuitively, Lemma 2 says that, in order to guarantee the feasibility of the final assignment, for any $j \in J^{(I)}$, it suffices to scale up $\sum_{i \in I} x^*_{i,j} + \sum_{i \in G} x'^{(\text{II})}_{i,j}$ by a factor at most two. Similar argument applies to any $j \in J^{(\leftrightarrow)}$, too. However, as scaling up $\sum_{i \in U} x'^{(0)}_{i,j}$ may not always be possible, we will instead guarantee that all the outlier clients are fully-assigned in the second stage. Note that this will provide extra amount of assignment needed to ensure the feasibility of j.

To be precise, for any $j \in \mathcal{D} \cup H$, define the scaling factor t'_j as follows. If $j \in \mathcal{D}$, i.e., j is a normal client, and $\sum_{i \in I} x^*_{i,j} + \sum_{i \in G} x'^{(\text{II})}_{i,j} > 0$, then

$$t'_j \;:=\; \frac{1}{\sum_{i \in I} x^*_{i,j} + \sum_{i \in G} x'^{(\text{II})}_{i,j}} \cdot \left(1 - \sum_{i \in U} x'^{(0)}_{i,j} - r'_j \right),$$

where we recall that r'_j is the amount of demand of j that is redistributed as outlier clients for any $j \in J^{(\leftrightarrow)}$. In the remaining cases, we define $t'_\ell := 1$.

Intuitively, t'_j is the factor for which $\sum_{i \in I} x^*_{i,j} + \sum_{i \in G} x'^{(\text{II})}_{i,j}$ should be scaled up in order for the client j to be fully-assigned. By the definition of t'_j and Lemma 2, we obtain the following corollary.

▶ **Corollary 3.** $0 \le t'_j \le 2$ *for all* $j \in \mathcal{D}$.

3.2 The Second Stage of the Rounding Process

In the second stage, the algorithm formulates the rounding decisions left for the outlier clusters, i.e., clusters centered at outlier clients in H, as a global optimization problem.

In particular, we formulate the rounding problem as a new instance of CFL-UFC with facility set $G := \bigcup_{j \in H} B(j)$ and client set U. Each $w \in U$ is associated with a demand d_w, defined as

$$d_w := \sum_{\substack{k \in H, \\ k \text{ located at } w}} \; \sum_{\substack{i \in B(k), \\ \ell \in \mathcal{D} \cup H}} t'_\ell \cdot x'^{(\text{II})}_{i,\ell}.$$

Intuitively, in the above definition, for each large facility $w \in U$, we consider all the outlier clusters that are centered at some outlier client located at w, and collect all the assignments within these clusters to be the demand of w. As described in the previous section, we scale up these assignment accordingly by the factor t'_ℓ for each client ℓ to meet the final feasibility.

We formulate the above instance as a carefully designed assignment LP, denoted LP-(O) and listed below. Our algorithm solves LP-(O) for a *basic optimal solution* $(\boldsymbol{x}'', \boldsymbol{y}'')$.

LP-(O)

$$\min \sum_{i \in G} y_i + \sum_{i \in G,\, j \in U} c_{i,j} \cdot x_{i,j}$$

$$\sum_{i \in G} x_{i,j} = d_j, \qquad \forall j \in U,$$

$$\sum_{j \in U} x_{i,j} \le u_i \cdot y_i, \qquad \forall i \in G,$$

$$y_i \le 1, \qquad \forall i \in G,$$

$$x_{i,j},\ y_i \ge 0, \qquad \forall i \in G,\ j \in U.$$

Properties Guaranteed in the 2^{nd}-stage

First we show that the feasible region of LP-(O) is nonempty, and the basic optimal solution $(\boldsymbol{x}'', \boldsymbol{y}'')$ can be computed. For any $w \in U$, let $H(w)$ denote the set of outlier clients located at w.

For any $w \in U$ and $i \in G$ such that $i \in B(k)$ for some $k \in H(w)$, i.e., i belongs to the clusters centered at some $k \in H(w)$, consider the bundled assignment $g_{i,w}$, defined as

$$g_{i,w} := \sum_{\ell \in \mathcal{D} \cup H} t'_\ell \cdot x'^{(\mathrm{II})}_{i,\ell}.$$

The following lemma is straightforward to verify.

▶ **Lemma 4.** $\left(\boldsymbol{g},\ 2\boldsymbol{y}'^{(II)} \right)$ *is a feasible solution for LP-(O).*

One of the key properties LP-(O) provides is that, basic feasible solutions of this LP provide a matching from the set of non-extremal facilities, i.e., $i \in G$ with $0 < y''_i < 1$, to the set of facilities in U, and hence an *unconditional roundup* can be performed on $\boldsymbol{y}''$ to obtain an integral solution. The following lemma is proved by explicitly considering the rank of the coefficient matrix.

▶ **Lemma 5.**

$$|L| \le |U|, \quad \textit{where } L := \left\{ i \in G :\ 0 < y''_i < 1 \right\}.$$

3.3 Final Output

When the rounding process ends, the algorithm defines the integral multiplicity function $\boldsymbol{y}^*$ as

$$y_i^* := \begin{cases} 1, & \text{if } i \in U \text{ or } i \in F_{D'}^*, \\ \lceil y_i'' \rceil, & \text{if } i \in G, \\ 0, & \text{otherwise.} \end{cases}$$

In particular, in addition to the large facilities in U and the facilities rounded up during the first stage of the process, the algorithm also performs an *unconditional roundup* on $\boldsymbol{y}''$ for the facilities in G. Then it solves the min-cost assignment problem on $\mathcal{D}$ and $\mathcal{F}^* := \{i \in \mathcal{F} : \ y_i^* = 1\}$ for an optimal integral assignment $\boldsymbol{x}^\dagger$, and outputs $(\boldsymbol{x}^\dagger, \boldsymbol{y}^*)$ as the approximation solution.

4 The Analysis

Let $\mathcal{A}$ denote the rounding algorithm in Section 3. We prove the following theorem.

▶ **Theorem 6.** *Let Ψ be an instance of CFL-UFC and $(\boldsymbol{x}', \boldsymbol{y}')$ be optimal for LP-(N) on Ψ. The rounding algorithm $\mathcal{A}$ computes in polynomial time a feasible integral solution $(x^\dagger, y^*)$ for Ψ with $\psi(x^\dagger, y^*) \le 4 \cdot \psi(x', y')$.*

The proof is outlined as follows. In Section 4.1, We define an assignment function $\boldsymbol{x}^\circ$ and shows that $(\boldsymbol{x}^\circ, y^*)$ is feasible for LP-(N) on Ψ. This shows that the feasible region of the min-cost assignment problem for $(\mathcal{D}, \mathcal{F}^*)$ is nonempty, and hence the integral assignment $\boldsymbol{x}^\dagger$ can be computed. In Section 4.2, we establish the 4-approximation guarantee for $(\boldsymbol{x}^\circ, y^*)$. This completes the proof for Theorem 6 since $\boldsymbol{x}^\dagger$ is the optimal solution for $(\mathcal{D}, \mathcal{F}^*)$.

Notations used in the analysis

In the following, we define notations and notions to describe our rounding process precisely in the analysis.

Consider the cluster-forming procedure. Let $\mathcal{C}_{D'}$ and $\mathcal{C}_{H'}$ denote the sets of clusters centered at the non-outlier clients and outlier clients, respectively. Recall that $F_{D'}^*$ denotes the set of facilities that are selected and rounded up for the clusters in $\mathcal{C}_{D'}$. Note that, $F_{D'}^* \cap G = \emptyset$, and the set of satellite facilities $B(j)$ for each $j \in H$ forms a partition of G.

For each outlier client $j \in H$, we use $w(j)$ to denote the facility in U at which j is located. We use $p(j)$ to denote the specific parent client in $J^{(\leftrightarrow)}$ from which j is created. On the contrary, for any $j \in J^{(\leftrightarrow)}$, we use $H(j)$ to denote the set of outlier clients that are created from j. For each $w \in U$, we use $H(w)$ to denote the set of outlier clients located at w.

Recall that, we use $(\boldsymbol{x}'^{(0)}, \boldsymbol{y}'^{(0)})$ to denote the initial solution the algorithm has for Ψ. For outlier clients $j \in H$ and any $i \in \mathcal{F}$, we use $\boldsymbol{x}'^{(0)}_{i,j}$ to denote the assignment made for j to i at the moment when j is created. We use $(\boldsymbol{x}'^{(\mathrm{II})}, \boldsymbol{y}'^{(\mathrm{II})})$ to denote the pair $(\boldsymbol{x}', \boldsymbol{y}')$ the algorithm maintains when it enters the second stage.

4.1 Feasibility of the Algorithm

In this section we define an intermediate assignment $\boldsymbol{x}^\circ$ and show that $(\boldsymbol{x}^\circ, \boldsymbol{y}^*)$ is a feasible solution for LP-(N) on the input instance Ψ.

Recall that, for any $w \in U$ and $i \in G$ such that $i \in B(k)$ for some $k \in H(w)$, we define the bundled assignment $g_{i,w}$ as $g_{i,w} := \sum_{\ell \in \mathcal{D} \cup H} t'_\ell \cdot x'^{(\mathrm{II})}_{i,\ell}$. Consider the basic optimal solution $(\boldsymbol{x}'', \boldsymbol{y}'')$ for LP-(O). For each $i \in G$ and $j \in \mathcal{D} \cup H$, we define the unbundled assignment h for the original clients j as

$$h_{i,j} \;:=\; \sum_{w \in U} x''_{i,w} \cdot \frac{1}{d_w} \cdot \sum_{k \in H(w),\, i' \in B(k)} t'_j \cdot x'^{(\mathrm{II})}_{i',j}.$$

Intuitively, in h we redistribute the assignment $\boldsymbol{x}''$ back for the original clients in $\mathcal{D} \cup H$ in a proportional way. It follows that for any $j \in \mathcal{D} \cup H$,

$$\begin{aligned}\sum_{i \in G} h_{i,j} \;&=\; \sum_{i \in G,\, w \in U} x''_{i,w} \cdot \frac{1}{d_w} \cdot \sum_{k \in H(w),\, i' \in B(k)} t'_j \cdot x'^{(\mathrm{II})}_{i',j} \\ &=\; \sum_{w \in U} \sum_{k \in H(w),\, i' \in B(k)} t'_j \cdot x'^{(\mathrm{II})}_{i',j} \;=\; \sum_{i \in G} t'_j \cdot x'^{(\mathrm{II})}_{i,j}, \end{aligned} \tag{1}$$

where in the second equality we apply the first constraint of LP-(O) and in the last equality we use the fact that the set of satellite facilities for each $j \in H$ forms a partition of G.

The Assignment $\boldsymbol{x}^\circ$

Provided the above, the assignment $\boldsymbol{x}^\circ$ for each $j \in \mathcal{D}$ is defined as

$$x^\circ_{i,j} \;:=\; \begin{cases} x'^{(0)}_{i,j}, & \text{if } i \in U, \\ t'_j \cdot x^*_{i,j} \;+\; \sum_{k \in H(j)} x^*_{i,k}, & \text{if } i \in F^*_{D'}, \\ h_{i,j} \;+\; \sum_{k \in H(j)} h_{i,k}, & \text{if } i \in G, \\ 0, & \text{otherwise.} \end{cases}$$

Intuitively, the assignment of each $j \in \mathcal{D}$ in $\boldsymbol{x}^\circ$ consists of its original assignments to U and the rounded assignments for clients in $\{j\} \cup H(j)$ to facilities in $F^*_{D'} \cup G$.

The following lemma is straightforward to verify.

▶ **Lemma 7.** *$(\boldsymbol{x}^\circ, \boldsymbol{y}^*)$ is feasible for LP-(N) on the input instance Ψ.*

4.2 Approximation Guarantee

In this section we establish the 4-approximation guarantee for $(\boldsymbol{x}^\circ, \boldsymbol{y}^*)$. First, we consider the cost incurred by clusters in $\mathcal{C}_{H'}$ and $\mathcal{C}_{D'}$ separately. Then we establish the overall guarantee.

Recall that, we use $p(j)$ for $j \in H$ to denote the client in $\mathcal{D}$ from which j is created. We extend the definition and define $p(k) := k$ for any $k \in \mathcal{D}$ for notational convenience.

Moreover, for any assignment $\boldsymbol{x}$ of interest, we will use $\boldsymbol{x}|_{A,B}$ to denote the assignments made in $\boldsymbol{x}$ between $A \subseteq \mathcal{F}$ and $B \subseteq \mathcal{D} \cup H$. Similarly, for any multiplicity function $\boldsymbol{y}$ of interest, we will use $\boldsymbol{y}|_A$ to denote the multiplicity of facilities in $A \subseteq \mathcal{F}$ in $\boldsymbol{y}$.

4.2.1 The clusters in $\mathcal{C}_{H'}$

The following lemma, which regards the assignment radius of the outlier clients in H, follows directly from the construction and triangle inequality.

▶ **Lemma 8.** *For any $j \in H$ and $i \in G$ such that $x'^{(II)}_{i,j} > 0$, we have $c_{i,j} \;\le\; \alpha_j$.*

In the following, we first bound the overall assignment cost in $\boldsymbol{x}^\circ|_{G,\mathcal{D}}$ in terms of that in $\boldsymbol{x}''$ and $\boldsymbol{x}'^{(\mathrm{II})}|_{G,\mathcal{D}\cup H}$. To this end, for any client $j \in \mathcal{D} \cup H$ and any $i \in G$, we need to bound the distance between i and $p(j)$. Let $w \in U$, $k \in H(w)$, and $i' \in B(k)$ is a satellite facility of k such that $x'^{(\mathrm{II})}_{i',j} > 0$. By the triangle inequality, we have

$$c_{i,p(j)} \;\le\; c_{i,w} \;+\; c_{i',w} \;+\; c_{i',p(j)} \;\le\; c_{i,w} \;+\; \alpha_k \;+\; c_{i',p(j)}, \tag{2}$$

where in the last inequality we apply Lemma 8. Also see Figure 2 for an illustration.

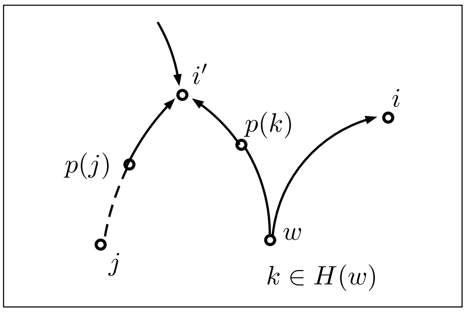

Figure 2 An illustration on the bundled assignment from $w \in U$ to $i \in G$ and unbundled assignments for $k \in H(w)$, $i' \in B(k)$ such that $x'^{(\mathrm{II})}_{i',j} > 0$.

By (2), we obtain the following lemma, which bounds the overall assignment cost in $\boldsymbol{x}^\circ|_{G,\mathcal{D}}$ in terms of that in $\boldsymbol{x}''$ and $\boldsymbol{x}'^{(\mathrm{II})}|_{G,\mathcal{D}\cup H}$.

▶ **Lemma 9.**

$$\sum_{i\in G,\, j\in\mathcal{D}} c_{i,j}\cdot x^\circ_{i,j} \;\le\; \sum_{i\in G,\, j\in U} c_{i,j}\cdot x''_{i,j} + \sum_{i\in G}\;\sum_{j\in\mathcal{D}\cup H} t'_j\cdot\left(\, c_{i,p(j)} + \alpha_j \,\right)\cdot x'^{(II)}_{i,j}.$$

In the following lemma, we expand $\boldsymbol{x}''$ and bound the overall cost incurred by $\boldsymbol{y}^*|_G$ and $\boldsymbol{x}''$ by the cost of $\boldsymbol{y}'^{(\mathrm{II})}|_G$, $\boldsymbol{y}'^{(0)}|_U$, and $\boldsymbol{x}'^{(\mathrm{II})}|_{G,\mathcal{D}\cup H}$.

▶ **Lemma 10.** *We have*

$$\sum_{i\in G}\lceil y''_i\rceil \;+\; \sum_{i\in G,\, j\in U} c_{i,j}\cdot x''_{i,j} \;\le\; 2\cdot\sum_{i\in G} y'^{(II)}_i \;+\; |L| \;+ \sum_{i\in G}\;\sum_{j\in\mathcal{D}\cup H} t'_j\cdot\alpha_j\cdot x'^{(II)}_{i,j},$$

where $L := \left\{\; i\in G \,:\; 0 < y''_i < 1 \;\right\}$.

Applying Lemma 9, Lemma 10, Lemma 5, and the fact that $y'^{(0)}_i \ge 1/2$ for all $i \in U$, we obtain the following bound for the cost incurred by $\left(\boldsymbol{x}^\circ|_{G,\mathcal{D}}\,,\,\boldsymbol{y}^*|_G\right)$.

$$\begin{aligned}
\sum_{i\in G} y^*_i \;+\; \sum_{i\in G,\, j\in\mathcal{D}} c_{i,j}\cdot x^\circ_{i,j} \quad&\le\quad 2\cdot\sum_{i\in G} y'^{(\mathrm{II})}_i \;+\; \sum_{i\in G,\, j\in H} t'_j\cdot\left(\, c_{i,p(j)} + 2\cdot\alpha_j \,\right)\cdot x'^{(\mathrm{II})}_{i,j} \\
&\qquad +\; 2\cdot\sum_{i\in U} y'^{(0)}_i \;+\; \sum_{i\in G,\, j\in\mathcal{D}} t'_j\cdot\left(\, c_{i,j} + 2\cdot\alpha_j \,\right)\cdot x'^{(\mathrm{II})}_{i,j} \\
&\le\quad 2\cdot\sum_{i\in G} y'^{(\mathrm{II})}_i \;+\; \sum_{i\in G,\, j\in H}\left(\, c_{i,p(j)} + 2\cdot\alpha_j \,\right)\cdot x'^{(\mathrm{II})}_{i,j} \\
&\qquad +\; 2\cdot\sum_{i\in U} y'^{(0)}_i \;+\; \sum_{i\in G,\, j\in\mathcal{D}}\left(\, 2\cdot c_{i,j} + 2\cdot t'_j\cdot\alpha_j \,\right)\cdot x'^{(\mathrm{II})}_{i,j},
\end{aligned}\tag{3}$$

where in the last inequality we apply Corollary 3 the fact that $t'_j \le 2$ for all $j \in \mathcal{D}$ and the definition that $t'_j = 1$ for all $j \in H$.

4.2.2 The clusters in $\mathcal{C}_{D'}$

Consider the cost incurred by the clusters in $\mathcal{C}_{D'}$. The following lemma, which bounds the total cost incurred by $\boldsymbol{x}^{\circ}|_{F^*_{D'},\mathcal{D}}$ and $\boldsymbol{y}^*|_{F^*_{D'}}$, is obtained by considering the cost of each cluster in $\mathcal{C}_{D'}$.

▶ **Lemma 11.**

(i)
$$\sum_{i\in F^*_{D'}} y^*_i \;\le\; 2\cdot\sum_{i\in I\setminus G} y'^{(0)}_i \;+\; 2\cdot\sum_{i\in G}\Big(\,y'^{(0)}_i - y'^{(II)}_i\Big). \tag{4}$$

(ii)
$$\begin{aligned}\sum_{i\in F^*_{D'},\, j\in\mathcal{D}} c_{i,j}\cdot x^{\circ}_{i,j} \;\le\; &\sum_{i\in F^*_{D'},\, j\in\mathcal{D}} 2\cdot t'_j\cdot\alpha_j\cdot x^*_{i,j} \;+\; \sum_{i\in F^*_{D'},\, j\in H} 2\cdot\alpha_j\cdot x^*_{i,j}\\ &+\sum_{i\in G,\, j\in\mathcal{D}} 2\cdot c_{i,j}\cdot\left(x'^{(0)}_{i,j} - \sum_{\ell\in H(j)} x'^{(0)}_{i,\ell} - x'^{(II)}_{i,j}\right) + \sum_{i\in G,\, j\in H} c_{i,p(j)}\cdot\Big(x'^{(0)}_{i,j} - x'^{(II)}_{i,j}\Big)\\ &+\sum_{i\in I\setminus G,\, j\in\mathcal{D}} 2\cdot c_{i,j}\cdot\left(x'^{(0)}_{i,j} - \sum_{\ell\in H(j)} x'^{(0)}_{i,\ell}\right) + \sum_{i\in I\setminus G,\, j\in H} c_{i,p(j)}\cdot x'^{(0)}_{i,j}.\end{aligned} \tag{5}$$

4.2.3 The overall guarantee

Combining Inequality (3), Inequality (4), and Inequality (5), and the construction scheme of outlier clients, we obtain the following lemma.

▶ **Lemma 12.**

$$\begin{aligned}\psi(\boldsymbol{x}^{\circ},\boldsymbol{y}^*) \;\le\; &4\cdot\sum_{i\in U} y'^{(0)}_i \;+\; 3\cdot\sum_{i\in U,\, j\in\mathcal{D}} c_{i,j}\cdot x'^{(0)}_{i,j}\\ &+\; 2\cdot\sum_{i\in I} y'^{(0)}_i \;+\; 2\cdot\sum_{i\in I,\, j\in\mathcal{D}} c_{i,j}\cdot x'^{(0)}_{i,j} \;+\; \sum_{j\in\mathcal{D}} 2\cdot\left(1-\sum_{i\in U} x'^{(0)}_{i,j}\right)\cdot\alpha_j.\end{aligned}$$

The following lemma follows from complementary slackness between $(\boldsymbol{x}',\boldsymbol{y}')$ and $(\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\Gamma},\boldsymbol{\eta})$, and the fact that $0 < y'^{(0)}_i < 1$ for all $i\in I$.

▶ **Lemma 13.**

$$\sum_{j\in\mathcal{D}}\left(1-\sum_{i\in U} x'^{(0)}_{i,j}\right)\cdot\alpha_j \;\le\; \sum_{i\in I} y'^{(0)}_i \;+\; \sum_{i\in I,\, j\in\mathcal{D}} c_{i,j}\cdot x'^{(0)}_{i,j}.$$

Applying Lemma 13 on Lemma 12, we obtain

$$\psi(\boldsymbol{x}^{\circ},\boldsymbol{y}^*) \;\le\; 4\cdot\sum_{i\in\mathcal{F}} y'^{(0)}_i \;+\; 4\cdot\sum_{i\in\mathcal{F},\, j\in\mathcal{D}} c_{i,j}\cdot x'^{(0)}_{i,j},$$

and Theorem 6 is proved.

References

1 Karen Aardal, Pieter L. van den Berg, Dion Gijswijt, and Shanfei Li. Approximation algorithms for hard capacitated k-facility location problems. *Eur. J. Oper. Res.*, 242(2):358–368, 2015. doi:10.1016/j.ejor.2014.10.011.

2 Hyung-Chan An, Mohit Singh, and Ola Svensson. Lp-based algorithms for capacitated facility location. In *55th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2014, Philadelphia, PA, USA, October 18-21, 2014*, pages 256–265. IEEE Computer Society, 2014. doi:10.1109/FOCS.2014.35.

3 Manisha Bansal, Naveen Garg, and Neelima Gupta. A 5-approximation for capacitated facility location. In Leah Epstein and Paolo Ferragina, editors, *Algorithms - ESA 2012 - 20th Annual European Symposium, Ljubljana, Slovenia, September 10-12, 2012. Proceedings*, volume 7501 of *Lecture Notes in Computer Science*, pages 133–144. Springer, 2012. doi:10.1007/978-3-642-33090-2_13.

4 Jarosław Byrka, Thomas Pensyl, Bartosz Rybicki, Aravind Srinivasan, and Khoa Trinh. An improved approximation for k-median and positive correlation in budgeted optimization. *ACM Trans. Algorithms*, 13(2), March 2017. doi:10.1145/2981561.

5 Kishen N. Gowda, Thomas W. Pensyl, Aravind Srinivasan, and Khoa Trinh. Improved bi-point rounding algorithms and a golden barrier for k-median. In Nikhil Bansal and Viswanath Nagarajan, editors, *Proceedings of the 2023 ACM-SIAM Symposium on Discrete Algorithms, SODA 2023, Florence, Italy, January 22-25, 2023*, pages 987–1011. SIAM, 2023. doi:10.1137/1.9781611977554.ch38.

6 Kamal Jain, Mohammad Mahdian, Evangelos Markakis, Amin Saberi, and Vijay V. Vazirani. Greedy facility location algorithms analyzed using dual fitting with factor-revealing lp. *J. ACM*, 50(6):795?824, November 2003. doi:10.1145/950620.950621.

7 Kamal Jain, Mohammad Mahdian, and Amin Saberi. A new greedy approach for facility location problems. In *Proceedings of the Thiry-Fourth Annual ACM Symposium on Theory of Computing*, STOC '02, pages 731–740, New York, NY, USA, 2002. Association for Computing Machinery. doi:10.1145/509907.510012.

8 Kamal Jain and Vijay V. Vazirani. Approximation algorithms for metric facility location and k-median problems using the primal-dual schema and lagrangian relaxation. *J. ACM*, 48(2):274–296, March 2001. doi:10.1145/375827.375845.

9 Mong-Jen Kao. Iterative partial rounding for vertex cover with hard capacities. In *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '17, pages 2638–2653, USA, 2017. Society for Industrial and Applied Mathematics.

10 Mong-Jen Kao. On the integrality gap of mfn relaxation for the capacitated facility location problem. In *Proceedings of the 2023 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1071–1089, 2023. doi:10.1137/1.9781611977554.ch40.

11 Retsef Levi, David B. Shmoys, and Chaitanya Swamy. Lp-based approximation algorithms for capacitated facility location. In George L. Nemhauser and Daniel Bienstock, editors, *Integer Programming and Combinatorial Optimization, 10th International IPCO Conference, New York, NY, USA, June 7-11, 2004, Proceedings*, volume 3064 of *Lecture Notes in Computer Science*, pages 206–218. Springer, 2004. doi:10.1007/978-3-540-25960-2_16.

12 Shi Li and Ola Svensson. Approximating k-median via pseudo-approximation. In *Proceedings of the Forty-Fifth Annual ACM Symposium on Theory of Computing*, STOC '13, page 901?910, New York, NY, USA, 2013. Association for Computing Machinery. doi:10.1145/2488608.2488723.

13 Mohammad Mahdian and Martin Pál. Universal facility location. In Giuseppe Di Battista and Uri Zwick, editors, *Algorithms - ESA 2003, 11th Annual European Symposium, Budapest, Hungary, September 16-19, 2003, Proceedings*, volume 2832 of *Lecture Notes in Computer Science*, pages 409–421. Springer, 2003. doi:10.1007/978-3-540-39658-1_38.

14 M. Pál, É. Tardos, and T. Wexler. Facility location with nonuniform hard capacities. In *Proceedings of the 42nd IEEE Symposium on Foundations of Computer Science*, FOCS '01, page 329, USA, 2001. IEEE Computer Society.

15 David B. Shmoys, Éva Tardos, and Karen Aardal. Approximation algorithms for facility location problems (extended abstract). In *Proceedings of the Twenty-Ninth Annual ACM Symposium on Theory of Computing*, STOC '97, pages 265–274, New York, NY, USA, 1997. Association for Computing Machinery. doi:10.1145/258533.258600.

16 Vincent Cohen-Addad Viallat, Fabrizio Grandoni, Euiwoong Lee, and Chris Schwiegelshohn. Breaching the 2 LMP approximation barrier for facility location with applications to k-median. In Nikhil Bansal and Viswanath Nagarajan, editors, *Proceedings of the 2023 ACM-SIAM Symposium on Discrete Algorithms, SODA 2023, Florence, Italy, January 22-25, 2023*, pages 940–986. SIAM, 2023. doi:10.1137/1.9781611977554.ch37.

17 David P. Williamson and David B. Shmoys. *The Design of Approximation Algorithms*. Cambridge University Press, USA, 1st edition, 2011.

18 Jiawei Zhang, Bo Chen, and Yinyu Ye. A multi-exchange local search algorithm for the capacitated facility location problem: (extended abstract). In George L. Nemhauser and Daniel Bienstock, editors, *Integer Programming and Combinatorial Optimization, 10th International IPCO Conference, New York, NY, USA, June 7-11, 2004, Proceedings*, volume 3064 of *Lecture Notes in Computer Science*, pages 219–233. Springer, 2004. doi:10.1007/978-3-540-25960-2_17.