Constant-Factor FPT Approximation for Capacitated k-Median

Marek Adamczyk
University of Warsaw, Poland
marek.adamczyk@mimuw.edu.pl

Jarosław Byrka
University of Wrocław, Poland
jby@cs.uni.wroc.pl

Jan Marcinkowski
University of Wrocław, Poland
jan.marcinkowski@cs.uni.wroc.pl

Syed M. Meesum
University of Wrocław, Poland
syedmohammad.meesum@uwr.edu.pl

Michał Włodarczyk
University of Warsaw, Poland
m.wlodarczyk@mimuw.edu.pl

Abstract

Capacitated k-median is one of the few outstanding optimization problems for which the existence of a polynomial time constant factor approximation algorithm remains an open problem. In a series of recent papers algorithms producing solutions violating either the number of facilities or the capacity by a multiplicative factor were obtained. However, to produce solutions without violations appears to be hard and potentially requires different algorithmic techniques. Notably, if parameterized by the number of facilities k, the problem is also $W[2]$ hard, making the existence of an exact FPT algorithm unlikely. In this work we provide an FPT-time constant factor approximation algorithm preserving both cardinality and capacity of the facilities. The algorithm runs in time $2^{\mathcal{O}(k \log k)} n^{\mathcal{O}(1)}$ and achieves an approximation ratio of $7 + \varepsilon$.

2012 ACM Subject Classification Theory of computation → Facility location and clustering; Theory of computation → Fixed parameter tractability

Keywords and phrases K-median, Clustering, Approximation Algorithms, Fixed Parameter Tractability

Digital Object Identifier 10.4230/LIPIcs.ESA.2019.1

Funding *Jarosław Byrka*: Supported by the Polish National Science Centre grant 2015/18/E/ST6/00456.
Jan Marcinkowski: Supported by the Polish National Science Centre grant 2015/18/E/ST6/00456.
Syed M. Meesum: Supported by the Polish National Science Centre grant 2015/18/E/ST6/00456.

1 Introduction

For many years approximation algorithms and FPT algorithms were developed in parallel. Recently the two paradigms are being combined and provide intriguing discoveries in the intersection of the two worlds. It is particularly interesting in the case of problems for which we fail to make progress on improving the approximation ratios in polynomial time. An excellent example of such a combination is the FPT approximation algorithm for the k-Cut problem by Gupta et al. [17].

In this work we focus on the Capacitated k-Median problem, whose approximability attracted attention of many researchers. Unlike in the case of the k-Cut problem, it is still not clear what approximation is possible for Capacitated k-Median in polynomial time. As shall be discussed in more detail in the following section, the best true approximation known is $\mathcal{O}(\log k)$ based on tree embedding of the underlying metric. The other algorithms either violate the bound on the number of facilities or the capacity constraints.

Our main result is a $(7+\epsilon)$-approximation algorithm for the Capacitated k-Median problem running in FPT(k) time, that exploits techniques from both – approximation and FPT – realms. The algorithm builds on the idea of clustering the clients into $\ell = \mathcal{O}(k \cdot (\log n)/\varepsilon)$ locations, which is similar to the approach from the $\mathcal{O}(\log k)$-approximation algorithm, where one creates $\mathcal{O}(k)$ clusters. This is followed by guessing the distribution of the k facilities inside these ℓ clusters. Having such a structure revealed, we simplify the instance further by rounding particular distances and reduce the problem to linear programming over a totally unimodular matrix.

1.1 Problems overview and previous work

In the Capacitated k-Median problem (CKM), we are given a set F of facilities, each facility f with a capacity $u_f \in \mathbb{Z}_{\geqslant 0}$, a set C of clients, a metric d over $F \cup C$ and an upper bound k on the number of facilities we can open. A solution to the CKM problem is a set $S \subseteq F$ of at most k open facilities and a connection assignment $\phi : C \to S$ of clients to open facilities such that $\left|\phi^{-1}(f)\right| \leqslant u_f$ for every facility $f \in S$. The goal of the problem is to find a solution that minimizes the connection cost $\sum_{c \in C} d(c, \phi(c))$. In the case when all the facilities can serve at most u clients, for some integer u, we obtain the Uniform CKM problem.

Uncapacitated k-median. The standard k-median problem, where there is no restriction on the number of clients served by a facility, can be approximated up to a constant factor [9, 2]. The current best is the $(2.675+\epsilon)$-approximation algorithm of Byrka et al. [4], which is a result of optimizing a part of the algorithm by Li and Svensson [23].

Approximability of CKM. As already stressed, Capacitated k-Median is among few remaining fundamental optimization problems for which it is not clear if there exist polynomial time constant factor approximation algorithms. All the known algorithms violate either the number of facilities or the capacities. In particular, already the algorithm of Charikar et al. [9] gave 16-approximate solution for the uniform capacitated k-median violating the capacities by a factor of 3. Then Chuzhoy and Rabani [10] considered general capacities and gave a 50-approximation algorithm violating capacities by a factor of 40.

The difficulty appears to be related to the unbounded integrality gap of the standard LP relaxation. To obtain integral solutions that are bounded with respect to the fractional solution to the standard LP, one has to either allow the integral solution to open twice as many facilities or to violate the capacities by a factor of two. LP-rounding algorithms essentially matching these limits have been obtained [1, 3].

Subsequently, Li broke this integrality gap barrier by giving a constant factor algorithm for the capacitated k-median by opening $(1+\varepsilon) \cdot k$ facilities [21, 22]. Afterwards analogous results, but violating the capacities by a factor of $(1+\varepsilon)$ were also obtained [5, 14].

The algorithms with $(1+\varepsilon)$ violations are all based on strong LP relaxations containing additional constraints for subsets of facilities. Notably, it is not clear if these relaxations can be solved exactly in polynomial time, still they suffice to construct an approximation algorithm via the "round-or-separate" technique that iteratively adds consistency constraints for selected subsets. Although while spectacularly breaking the standard LP integrality bound, these techniques appear insufficient to yield a proper approximation algorithm that does not violate constraints.

The only true approximation for CKM known is a folklore $\mathcal{O}(\log k)$ approximation algorithm that can be obtained via the metric tree embedding with expected logarithmic distortion [15]. To the best of our knowledge, this result has not been explicitly published, but it can be obtained similarly to the $\mathcal{O}(\log k)$-approximation for Uncapacitated KM by Charikar [7]. For the sake of completeness and since it follows easily from our framework, we give its proof in Section 3 without claiming credit for it. This $\mathcal{O}(\log k)$ barrier is in contrast with other capacitated clustering problems such as facility location and k-center, for which constant factor approximation algorithms are known [19, 12].

After our work was announced, Xu et al. [27] proposed a similar algorithm for Euclidean Capacitated k-Means, i.e., a constant factor approximation running in time FPT(k). Both our and their approximation ratios have been very recently improved by Cohen-Addad and Li [11], who obtained $(3+\varepsilon)$ for Capacitated k-median and $(9+\varepsilon)$ for Capacitated k-means in general metric spaces. They have also provided a deeper insight into the problems basing on the framework of coresets.

1.2 Parameterized Complexity

A parameterized problem instance is created by associating an input instance with an integer parameter k. We say that a problem is *fixed parameter tractable* (FPT) if every instance (I, k) of the problem can be solved in time $f(k) \cdot |I|^{\mathcal{O}(1)}$, where f is an arbitrary computable function of k.

We say that a problem is FPT if it is possible to give an algorithm that solves it in running time of the required form. Such an algorithm we shall call a *parameterized algorithm*.

To show that a problem is unlikely to be FPT, we use parameterized reductions analogous to those employed in the classical complexity theory. Here, the concept of W-hardness replaces the one of NP-hardness, and we need not only construct an equivalent instance in FPT time, but also ensure that the size of the parameter in the new instance depends only on the size of the parameter in the original instance. In contrast to the NP-hardness theory, there is a hierarchy of classes FPT = W[0] $\subseteq$ W[1] $\subseteq$ W[2] $\subseteq \ldots$ and these containments are believed to be strict. If there exists a parameterized reduction transforming a problem known to be W[t]-hard for $t > 0$ to another problem Π, then the problem Π is W[t]-hard as well. This provides an argument that Π is unlikely to admit an algorithm with running time $f(k) \cdot |I|^{\mathcal{O}(1)}$.

We begin with an argument that allowing FPT time for (even uncapacitated) k-Median should not help in finding the optimal solution and we still need to settle for approximation.

▷ Fact 1. The Uncapacitated k-Median problem is W[2]-hard when parameterized by k, even on metrics induced by unweighted graphs.

Proof. Consider an instance (G, k) of the Dominating Set problem, which is W[2]-hard when parameterized by the solution size k. Graph G induces a metric such that the distance between two adjacent vertices equals one and otherwise the distance between vertices is the length of the shortest path. A dominating set of size at most k exists in graph G if and only if we can find a vertex set S of size k, such that all the other vertices are at distance 1 from S. This is equivalent to the solution to Uncapacitated k-Median on the metric induced by G being of size exactly $|V(G)| - k$. ◁

Parameterized Approximation

In recent years new research directions emerged in the intersection of the theory of approximation algorithms and the FPT theory. It turned out that for some problems that are intractable in the exact sense, parameterization still comes in useful when we want to reduce the approximation ratio. Some examples are $(2-\varepsilon)$-approximation for k-Cut [17] or $f(\mathcal{F})$-approximation for Planar-$\mathcal{F}$ Deletion [16] for some implicit function f. The dependency on $\mathcal{F}$ was later improved, leading to $\mathcal{O}(\log k)$-approximations for, e.g., k-Vertex Separator[20] and k-Treewidth Deletion [18].

On the other hand some problems parameterized by the solution size have been proven resistant to such improvements. Chalermsook et al. [6] observed that under the assumption of Gap-ETH there can be no parameterized approximation with ratio $o(k)$ for k-Clique and none with ratio $f(k)$ for k-Dominating Set (for any function f). Subsequently Gap-ETH has been replaced with a better established hardness assumption FPT $\neq$ W[1] for k-Dominating Set [25].

1.3 Organization of the paper

Our main result is stated in Theorem 16 (Section 4.3), where we present a $(7+\epsilon)$-approximation algorithm for the Non-Uniform CKM problem running in FPT(k) time.

To obtain this result we need two ingredients. The first one is a metric embedding that reduces the problem to a simpler instance, called ℓ-centered, which is described in Section 2. This reduction provides a richer structure, which can be exploited to obtain the folklore $\mathcal{O}(\log k)$-approximation via tree embeddings [15]. As already mentioned, similar approach was presented by Charikar et al. [7] in their algorithm for the uncapacitated setting. For the sake of completeness, we present this result in Appendix 3.

The second ingredient is a parameterized algorithm for the ℓ-centered instances. Since it is simpler in the uniform setting, we first solve it in Section 4.2 as a warm up before the main result. This way the new ideas are being revealed gradually to the reader.

2 ℓ-Centered instances

Suppose we work with a graph on nodes $F \cup C$, on which we are given a metric d. In our considerations the set $F \cup C$ will be fixed throughout, however we will be modifying the metric over it. Consider an algorithm ALG which produces a solution $ALG(d)$ for a metric d. This solution can be seen as a mapping which we explicitly denote by $\phi^{ALG(d)}$. Its cost in the metric d' equals $\sum_{c\in C} d'\left(c, \phi^{ALG(d)}\right)$ which we shall briefly denote by $cost\left(\phi^{ALG(d)}, d'\right)$. The second argument is useful, when an algorithm ALG produces a solution (mapping) $ALG(d)$ with respect to metric d, but later on we may be interested in its cost over a different metric. Also, let $OPT(d)$ denote the optimum solution for the CKM problem on metric d.

In order to solve CKM, we shall invoke an algorithm for Uncapacitated KM as a subroutine. Let $ALG^{\ell}_{unc}(d)$ be a relaxed solution that opens up to $\ell \geqslant k$ facilities and can break the capacity constraints. It induces a mapping which, for consistency, we shall denote by $\phi^{ALG^{\ell}_{unc}(d)}$. Observe that in this mapping every client can be connected to the closest open facility. Since Uncapacitated KM admits constant approximation algorithms, we can work with solutions satisfying: $cost\left(\phi^{ALG^{\ell}_{unc}(d)}, d\right) = O\left(cost\left(\phi^{OPT(d)}, d\right)\right)$. The larger ℓ we allow in the relaxation, the smaller constant we will be able to achieve in the relation above.

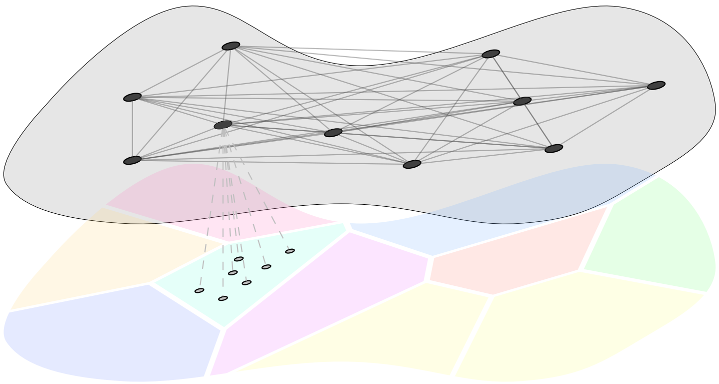

Figure 1 An ℓ-centered instance. In the upper layer there is a set S of ℓ vertices connected as a clique. The rest of vertices are divided into separate clusters. Vertices in a single cluster are only connected to their center in the set S.

Using such an algorithm for Uncapacitated KM as a subroutine, we can find a simpler metric to work with. First we build a graph which will induce the metric. Let $F(ALG^{\ell}_{unc}(d))$ be the set of facilities opened by $ALG^{\ell}_{unc}(d)$. For each such a facility f we create a copy vertex s^f, which is at distance 0 from f. We denote the set of copies by S, i.e., $S = \left\{ s^f \,\middle|\, f \in F(ALG^{\ell}_{unc}(d)) \right\}$. Given that we demand the distance from f to s^f to be 0, we can naturally extend the metric d to the set $C \cup F \cup S$. To distinguish facilities from $F(ALG^{\ell}_{unc}(d))$ from their copies S, we shall call each copy $s \in S$ a *center*.

We build a complete graph on S and preserve the metric d therein. For every node $v \notin S$, be it either a client from C or a facility from F, we place an edge to the closest (according to the extended d) center $s^v \in S$ and set its length to $d(v, s^v)$. We call such a graph ℓ-centered and refer to its induced metric as d_ℓ.

▶ **Definition 2.** *An instance of* CKM *is called* ℓ-centered *if the metric, which we shall denote by d_ℓ, is induced by a weighted graph $G(F \cup C \cup S, E)$ such that*
1. $|S| \leqslant \ell$,
2. $\binom{S}{2} \subseteq E$*, i.e., S forms a clique,*
3. *for every $v \in C \cup F$ there is only one edge incident to v in E, and it connects v to some $s^v \in S$.*

For a center $s \in S$ we shall say that all nodes from $F \cup C$ that are connected to s form a *cluster* of s. If we consider only nodes from F, then we talk about an f-cluster of s, denoted $F(s)$.

The idea of preprocessing that simplifies the metric by recognizing a small number of hubs resembles the notion of α-preserving metrics, that have been used as a tool to obtain coresets for the related problem Balanced k-Median [13].

In the following lemma we relate the cost of embedding the optimum solution $OPT(d)$ from a metric d to d_ℓ.

▶ **Lemma 3** (Embedding d into ℓ-centered metric d_ℓ)**.** *Let $ALG^{\ell}_{unc}(d)$ be a solution for the* Uncapacitated KM *problem on metric d from which we construct the ℓ-centered instance. Optimal solution $OPT(d)$ can be embedded into an ℓ-centered metric d_ℓ with the cost relation being*

$$cost\left(\phi^{OPT(d)}, d\right) \leqslant cost\left(\phi^{OPT(d)}, d_\ell\right) \leqslant 3 \cdot cost\left(\phi^{OPT(d)}, d\right) + 4 \cdot cost\left(\phi^{ALG^{\ell}_{unc}(d)}, d\right).$$

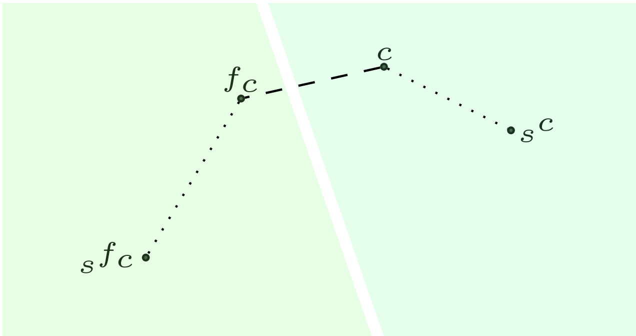

Figure 2 Situation in Lemma 3. In the optimum solution to the CKM instance, client c is connected to the facility f_c. In the ℓ-centered instance c resides in a cell, where s^c is a center. The center of f_c is s^{f_c}.

Proof. Let c be a client connected to facility f_c in the optimal solution $OPT(d)$. Let s^c be the center closest to c within S (the ℓ-center), and let s^{f_c} be the center closest to f_c. First let us note that $d_\ell(c, f_c) = d(c, s^c) + d\left(s^c, s^{f_c}\right) + d\left(s^{f_c}, f_c\right)$. Next we bound the terms $d\left(f_c, s^{f_c}\right)$ and $d\left(s^c, s^{f_c}\right)$ separately.

▷ Fact 4. For every client c and its facility f_c from OPT we have $d\left(f_c, s^{f_c}\right) \leqslant d(f_c, c) + d(c, s^c)$.

Proof. Since s^{f_c} is the closest ℓ-center to the facility f_c, we have that $d\left(f_c, s^{f_c}\right) \leqslant d(f_c, s^c)$. At the same time, from the triangle inequality it follows that $d(f_c, s^c) \leqslant d(f_c, c) + d(c, s^c)$. ◁

▷ Fact 5. For each c we have $d\left(s^c, s^{f_c}\right) \leqslant 2\left(d(f_c, c) + d(c, s^c)\right)$.

Proof. From the triangle inequality we know that

$$d\left(s^c, s^{f_c}\right) \leqslant d(s^c, c) + d(c, f_c) + d\left(f_c, s^{f_c}\right).$$

From Fact 4 we also know that $d\left(f_c, s^{f_c}\right) \leqslant d(f_c, c) + d(c, s^c)$, and combining the two inequalities we get $d\left(s^c, s^{f_c}\right) \leqslant d(s^c, c) + d(c, f_c) + d\left(f_c, s^{f_c}\right) \leqslant 2\left(d(f_c, c) + d(c, s^c)\right)$. ◁

These two facts imply

$$\begin{aligned} d_\ell(c, f_c) &= d(c, s^c) + d\left(s^c, s^{f_c}\right) + d\left(s^{f_c}, f_c\right) \\ &\leqslant d(c, s^c) + d\left(s^c, s^{f_c}\right) + \left(d(f_c, c) + d(c, s^c)\right) && \text{(from Fact 4)} \\ &\leqslant d(c, s^c) + 2\left(d(f_c, c) + d(c, s^c)\right) + \left(d(f_c, c) + d(c, s^c)\right) && \text{(from Fact 5)} \\ &= 3 \cdot d(f_c, c) + 4 \cdot d(c, s^c), \end{aligned}$$

which implies the second inequality from the statement of Lemma 3. The first one directly comes from the triangle inequality $d(c, f_c) \leqslant d(c, s^c) + d\left(s^c, s^{f_c}\right) + d\left(s^{f_c}, f_c\right) = d_\ell(c, f_c)$, completing the whole proof. ◀

The next lemma is quite simple. Its proof follows from the fact that metric d_ℓ dominates the metric d, i.e., $d_\ell(u, v) \geqslant d(u, v)$ for all pairs of vertices $u, v \in C \cup F$.

▶ **Lemma 6** (Going back from ℓ-centered metric d_ℓ to d). *Any solution for the ℓ-centered metric d_ℓ can be embedded back into d without any loss:*

$$cost\left(\phi^{ALG(d_\ell)}, d_\ell\right) \geqslant cost\left(\phi^{ALG(d_\ell)}, d\right).$$

Blending together Lemmas 3 and 6 we can state the following lemma about reducing the CKM problem to ℓ-centered instances.

▶ **Lemma 7.** *Suppose we are given a solution $ALG^{\ell}_{unc}(d)$ for the* Uncapacitated KM *problem on metric d which opens ℓ centers, but β-approximates the optimum solution $OPT^{k}_{unc}(d)$ for* Uncapacitated KM *problem with k centers, i.e., $cost\left(ALG^{\ell}_{unc}(d), d\right) \leqslant \beta \cdot cost\left(OPT^{k}_{unc}(d), d\right)$. Suppose we are given an α-approximation algorithm for the* CKM *problem on ℓ-centered instances. If so, then we can construct an $\alpha \cdot (3 + 4\beta)$-approximation algorithm for* CKM *on general instances.*

Proof. Suppose that we have an α-approximation solution for the ℓ-centered instance with metric d_ℓ, i.e., $ALG(d_\ell)$ such that

$$cost\left(\phi^{ALG(d_\ell)}, d_\ell\right) \leqslant \alpha \cdot cost\left(\phi^{OPT(d_\ell)}, d_\ell\right).$$

Since $OPT(d)$ is some solution for the ℓ-centered instance with metric d_ℓ we have

$$cost\left(\phi^{ALG(d_\ell)}, d_\ell\right) \leqslant \alpha \cdot cost\left(\phi^{OPT(d_\ell)}, d_\ell\right) \leqslant \alpha \cdot cost\left(\phi^{OPT(d)}, d_\ell\right).$$

And from Lemma 3 we have that

$$\begin{aligned} cost\left(\phi^{ALG(d_\ell)}, d_\ell\right) &\leqslant \alpha \cdot cost\left(\phi^{OPT(d_\ell)}, d_\ell\right) \\ &\leqslant \alpha \cdot cost\left(\phi^{OPT(d)}, d_\ell\right) \\ &\leqslant \alpha\left(3 \cdot cost\left(\phi^{OPT(d)}, d\right) + 4 \cdot cost\left(\phi^{ALG^{\ell}_{unc}(d)}, d\right)\right). \end{aligned}$$

Since solution $ALG^{\ell}_{unc}(d)$ β-approximates the optimal solution $OPT^{k}_{unc}(d)$ for Uncapacitated KM with k centers on metric d, we have that

$$cost\left(\phi^{ALG^{\ell}_{unc}(d)}, d\right) \leqslant \beta \cdot cost\left(\phi^{OPT^{k}_{unc}(d)}, d\right) \leqslant \beta \cdot cost\left(\phi^{OPT(d)}, d\right).$$

The second inequality $cost\left(\phi^{OPT^{k}_{unc}(d)}, d\right) \leqslant cost\left(\phi^{OPT(d)}, d\right)$ follows from an obvious fact that uncapacitated version of the problem is easier than the capacitated. Hence

$$\begin{aligned} cost\left(\phi^{ALG(d_\ell)}, d_\ell\right) &\leqslant \alpha\left(3 \cdot cost\left(\phi^{OPT(d)}, d\right) + 4 \cdot cost\left(\phi^{ALG^{\ell}_{unc}(d)}, d\right)\right) \\ &\leqslant \alpha\left(3 \cdot cost\left(\phi^{OPT(d)}, d\right) + 4\beta \cdot cost\left(\phi^{OPT(d)}, d\right)\right) \\ &\leqslant \alpha\,(3 + 4\beta) \cdot cost\left(\phi^{OPT(d)}, d\right). \end{aligned}$$

Since without any loss we can embed the solution $ALG(d_\ell)$ for the ℓ-centered metric d_ℓ into the initial metric d (Lemma 6) we obtain an $\alpha \cdot (3 + 4\beta)$-approximation algorithm. The claim follows. ◀

3 $\mathcal{O}(\log k)$-approximation in polynomial time

In this section we present a folklore polynomial-time $\mathcal{O}(\log k)$-approximation algorithm for CKM. Since constant-factor approximation algorithms for Uncapacitated KM exist [9], it is a clear consequence of Lemma 7 with β being constant that it is sufficient for us to construct an $\mathcal{O}(\log k)$-approximation algorithm for the k-centered instances.

A standard tool to provide such a guarantee is the *Probabilistic Tree Embedding* by [15]. This makes our algorithm a randomized one, but if needed, it is possible to derandomize it using the ideas from [8].

▶ **Definition 8.** *A set of metric spaces $\mathcal{T}$ together with a probability distribution $\pi_{\mathcal{T}}$ over $\mathcal{T}$ probabilistically α-approximates the metric space (X, d) if*

1. *Every metric $\tau \in \mathcal{T}$ dominates (X, d), that is, $d(x, y) \leqslant \tau(x, y)\ \forall x, y \in X$.*
2. *For every pair of points $x, y \in X$ its expected distance is not expanded by more then α, i.e.,*

$$\mathbb{E}_{\tau \sim \pi_{\mathcal{T}}}[\tau(x, y)] \leqslant \alpha \cdot d(x, y).$$

It is a well-known fact, that any metric (X, d), can be probabilistically $\mathcal{O}(\log |X|)$-approximated by a distribution of tree metrics, such that the points in X are the leaves in the resulting tree [15].

As described in Definition 2, our k-centered metric d_k is induced by a graph composed of two layers – the set S of k vertices connected in a clique, and the rest of vertices, $F \cup C$, each connected to only one vertex in S. Let T be a random tree embedding of the set S (with a metric function d_T). A modified instance G_T of our problem is created by replacing the clique S with its tree approximation T.

▶ **Lemma 9.** *An optimum solution for* CKM *on the instance G_T is in expectation at most $\mathcal{O}(\log k)$ times larger than the optimum for the metric d_k.*

Proof. $OPT(d_k)$ denotes the optimum mapping of clients to k facilities in the k-centered metric d_k. Consider client c and facility $f = \phi^{OPT(d_k)}(c)$. Let now s^c be the center of c and s^f the center of f. The cost of connecting client c to f amounts to

$$d_k(c, f) = d_k(c, s^c) + d_k\left(s^c, s^f\right) + d_k\left(s^f, f\right)$$

in the metric d_k.

The guarantee of tree embeddings gives us an upper bound on a cost of applying the same mapping in the instance G_T,

$$\begin{aligned} \mathbb{E}\left[d_T(c, f)\right] = & \quad d_k(c, s^c) + \mathbb{E}\left[d_T\left(s^c, s^f\right)\right] + d_k\left(s^f, f\right) \\ \leqslant & \quad d_k(c, s^c) + \mathcal{O}(\log k) \cdot d_k\left(s^c, s^f\right) + d_k\left(s^f, f\right) \\ \leqslant & \quad \mathcal{O}(\log k) \cdot d_k(c, f). \end{aligned}$$

Which means that $\mathbb{E}\left[cost\left(\phi^{OPT(d_k)}, d_T\right)\right] \leqslant \mathcal{O}(\log k) \cdot cost\left(\phi^{OPT(d_k)}, d_{G_k}\right)$. Moreover, $OPT(d_k)$ might not be the optimal solution for the metric d_T, yet its optimal solution can only have smaller cost:

$$cost\left(\phi^{OPT(d_T)}, d_T\right) \leqslant cost\left(\phi^{OPT(d_k)}, d_T\right)$$ ◀

▶ **Theorem 10.** *The CKM problem admits an $\mathcal{O}(\log k)$-approximation algorithm with polynomial running time.*

Proof. After applying the probabilistic tree embedding to the graph inducing d_k – as presented in Lemma 9 – we obtain a tree instance G_T. It should come as no surprise that the problem is polynomially solvable on trees and we explain how to find the optimum solution on G_T in Lemma 12. The assignment $\phi^{OPT(d_T)}$, which yields the minimum cost on the tree G_T, can be now used to match clients to facilities in the original instance. It does not incur any additional cost, as

$$cost\left(\phi^{OPT(d_T)}, d_T\right) \geqslant cost\left(\phi^{OPT(d_T)}, d_k\right) \geqslant cost\left(\phi^{OPT(d_T)}, d\right)$$

from the property (1) of Definition 8 and Lemma 6. Combining this with a bound on $\mathbb{E}\left[cost\left(\phi^{OPT(d_k)}, d_T\right)\right]$ from Lemma 9 finishes the proof. ◀

3.1 CKM on a tree

The second ingredient of the $\mathcal{O}(\log k)$-approximation for CKM is an exact algorithm solving the problem on trees. We will now describe a simple, polynomial-time procedure for this special case. In our algorithm we can assume, that all the clients and facilities reside in leaves, but the principle is easy to extend to the general problem on trees. We first turn the tree into a complete binary tree by adding dummy vertices and edges of length 0 (which may double its size).

Suppose we have a subtree t of the tree instance, hanging on an edge e_t. Once we have decided, which facilities to open inside the subtree t, we know if their total capacity is sufficient to serve all the clients inside t. If not, then we need to route some clients' connections to the facilities outside through the edge e_t. However, if the facilities we have opened in t have enough total capacity to serve some b clients from the outside, we will connect them through the edge e_t.

▶ **Definition 11.** *$D(t, k', b)$, for subtree t, number $k' \in \{0, \ldots, k\}$ of facilities and balance $b \in \{-n, \ldots, n\}$, is the minimum cost of opening exactly k' facilities in t and routing exactly b clients down through e_t ($b < 0$ would mean that we are routing $-b$ clients up). The cost of routing is counted to the top endpoint of e_t.*

▶ **Lemma 12.** *The* CKM *problem on trees admits a polynomial time exact algorithm.*

Proof. Computing $D(t, k', b)$ on t with two children t_1 and t_2 amounts to finding k'_1, k'_2, b_1 and b_2 that minimize

$$D(t_1, k'_1, b_1) + D(t_2, k'_2, b_2),$$

such that $b_1 + b_2 = b$ and $k'_1 + k'_2 = k'$. They can be trivially found in $\mathcal{O}(k \cdot n)$ time for a single pair $\langle k', b\rangle$. Once k'_1, k'_2, b_1 and b_2 are found, we set

$$D(t, k', b) = D(t_1, k'_1, b_1) + D(t_2, k'_2, b_2) + d(e_t) \cdot |b|,$$

where $d(e)$ is the length of the edge in our tree. For a leaf l, $D(l, k', b)$ is defined naturally, depending on whether the leaf holds a client or a facility. Note, that for a leaf with a facility, $D(l, 1, b)$ is finite also for b smaller than the capacity of the facility, as the optimal solution might not use it entirely. Finally, the optimum solution to the CKM problem on the entire tree T is equal to $\min_{k' \in \{1,\ldots,k\}} D(T, k', 0)$. ◀

4 Constant factor approximation

In this section we present the main result of the paper which is a $(7 + \varepsilon)$-approximation algorithm for the Non-Uniform CKM problem. We precede it with a $(7 + \varepsilon)$-approximation algorithm for the Uniform CKM problem to introduce the ideas gradually. Both algorithms enumerate configurations of open facilities' locations, and as a subroutine we need to use an algorithm which, for a fixed configuration of k open facilities, finds the optimal assignment of clients to facilities. This subroutine is presented in the following subsection.

4.1 Optimal mapping subroutine

We are given an ℓ-centered metric instance $(F \cup C \cup S, d_\ell)$ of the k-median problem. Suppose that we have already decided to open a fixed subset $F^{open} \subseteq F$ of the facilities assume $|F^{open}| \leqslant k$. and we look for a mapping $\phi : C \to F^{open}$. In the uncapacitated case we can just assign each client to the closest facility in F^{open}. It turns out that even in the capacitated setting we can find the mapping ϕ optimally in polynomial time for a given F^{open}. We state the problem of finding the optimal ϕ as an integer program:

$$\begin{aligned}
\text{minimize} \quad & \sum_{c\in C}\sum_{f\in F^{open}} d_\ell(c,f)\cdot x_{c,f} && \text{(MAPPING-IP)}\\
\text{subject to} \quad & \sum_{f\in F^{open}} x_{c,f} = 1 && \forall c\in C,\\
& \sum_{c\in C} x_{c,f} \leqslant u_f && \forall f\in F^{open},\\
& x_{c,f}\in\{0,1\}.
\end{aligned}$$

In the above program $x_{c,f} = 1$ represents the fact that $\phi(c) = f$.

▶ **Lemma 13.** *We can find an optimal solution to the* (MAPPING-IP) *in polynomial time.*

Proof. The proof follows from the fact that the relaxation of the above integer program – a program which differs from (MAPPING-IP) only with the $x_{c,f} \geqslant 0$ constraints instead of $x_{c,f} \in \{0,1\}$ – has an optimal solution which is integral. To see this, observe that the linear program is a formulation of the transportation problem. For such a linear program, the constraint matrix is totally unimodular, which implies the integrality of an extremal solution. See [26] for a reference. ◀

4.2 Uniform case

We begin with a parameterized algorithm for the uniform case. It is simpler than the general case, as knowing the number of facilities to open in f-cluster $F(s)$ allows us to choose them greedily.

▶ **Lemma 14.** Uniform CKM *can be solved exactly in time $\ell^k \cdot n^{\mathcal{O}(1)}$ on ℓ-centered instances.*

Proof. Let $(F\cup C\cup S, d_\ell)$ be the ℓ-centered metric. Note that the f-clusters partition the whole set of facilities, i.e., $\cup_{s\in S}F(s) = F$. Let $OPT\,(d_\ell)$ be an optimal solution for the CKM problem on d_ℓ. Every facility $f\in F$ belongs to exactly one f-cluster $F(s)$. Hence, the f-clusters partition the set of k facilities opened by $OPT(d_\ell)$. Let us look at all the facilities from a particular f-cluster $F(s)$ opened by $OPT(d_\ell)$, and suppose that $OPT(d_\ell)$ opens k_s of facilities in $F(s)$. Since we consider a uniform capacity case, we can assume without loss that these k_s open facilities from $F(s)$ are exactly the ones that are closest to s.

Therefore, if we know what is the number of facilities that $OPT(d_\ell)$ opens in each f-cluster, then we would know what the exact set of open facilities in $OPT\,(d_\ell)$ is due to the greediness in each f-cluster. To find out this allocation we can simply enumerate over all possibilities. We just need to scan over all *configurations* $(k_s)_{s\in S}$ where $\sum_s k_s = k$. Since there are k facilities to open, and each of them can belong to one of ℓ f-clusters $F(s)$, there are at most ℓ^k possible configurations. Of course some configurations may not be feasible since it may happen that $k_s > |F(s)|$, but these can be simply ignored.

For each configuration $(k_s)_{s\in S}$ we need to find the optimal mapping of clients to the set of open facilities that preserves their capacities. Let $F\left((k_s)_{s\in S}\right)$ be the set of open facilities induced by configuration $(k_s)_{s\in S}$, that is, where we greedily open k_s facilities in f-cluster $F(s)$. Given $F\left((k_s)_{s\in S}\right)$, to find the optimal mapping we use the polynomial time exact algorithm from Lemma 13 with $F^{open} = F\left((k_s)_{s\in S}\right)$.

Once we know the optimal assignment for each configuration, we can simply take the cheapest one, knowing that it is the optimal one. This proves the lemma. ◀

This lemma suffices to obtain a $(7+\varepsilon)$-approximation for Uniform CKM with a reasoning that we will present in Theorem 16 in full generality.

4.3 Non-uniform case

▶ **Lemma 15.** Non-Uniform CKM *can be solved with approximation ratio* $(1+\epsilon)$ *in time* $\left(\mathcal{O}\left(\ell \cdot \frac{1}{\varepsilon} \ln \frac{n}{\varepsilon}\right)\right)^k n^{\mathcal{O}(1)}$ *on ℓ-centered instances.*

Proof. We begin with guessing the largest distance in d_ℓ between a client and a facility that would appear in the optimal solution – let us denote this quantity as D. There are at most $\mathcal{O}(n^2)$ choices for D, and from now we assume that it is guessed correctly. Note that $D \leqslant cost\left(OPT(d_\ell), d_\ell\right)$ and $D \geqslant d(f, s_f)$ for all facilities opened by $OPT(d_\ell)$.

Consider the set of facilities $F(s)$ in the cluster of a center s. We can remove all facilities f such that $d(s,f) > D$, because they cannot be a part of the optimal solution. Let us partition remaining facilities from $F(s)$ into buckets $F_0(s), F_1(s), \ldots, F_{\lceil \log_{1+\varepsilon} \frac{n}{\varepsilon} \rceil}(s)$, such that

$$F_i(s) = \begin{cases} \left\{ f \in F(s) \,\middle|\, d(s,f) \in \left[(1+\varepsilon)^{-(i+1)} D, (1+\varepsilon)^{-i} D\right] \right\} & \text{for } i < \lceil \log_{1+\varepsilon} \frac{n}{\varepsilon} \rceil \\ \left\{ f \in F(s) \,\middle|\, d(s,f) \in \left[0, (1+\varepsilon)^{-\lceil \log_{1+\varepsilon} \frac{n}{\varepsilon} \rceil} D\right] \right\} & \text{for } i = \lceil \log_{1+\varepsilon} \frac{n}{\varepsilon} \rceil \end{cases}$$

The number of buckets equals $\log_{1+\varepsilon} \frac{n}{\varepsilon} = \frac{1}{\ln(1+\varepsilon)} \ln \frac{n}{\varepsilon} = \mathcal{O}\left(\frac{1}{\varepsilon} \ln \frac{n}{\varepsilon}\right)$. We modify the metric again by setting $d'_\ell(s,f) = (1+\varepsilon)^{-i} D$ for $f \in F_i(s)$. The distances within S remain untouched. Observe that the distances can only increase.

We shall guess the structure of the solution $OPT(d'_\ell)$ similarly as in Lemma 14. For each of the k facilities, we can choose its location as follows: first we choose one of the ℓ-centers s (ℓ choices), and then we choose one of the $F_i(s)$ partitions ($\mathcal{O}\left(\frac{1}{\varepsilon} \ln \frac{n}{\varepsilon}\right)$ choices). Let us denote the number of facilities in a particular partition $F_i(s)$ as $k_{s,i}$. We can assume that $k_{s,i} \leqslant |F_i(s)|$ because otherwise we know that the guess was incorrect. Since $d'_\ell(s,f)$ is the same for all $f \in F_i(s)$, we can assume the optimal solution opens $k_{s,i}$ facilities with the biggest capacities.

Once we establish the set of facilities to open, we can find the optimal assignment in metric d'_ℓ using the polynomial time exact subroutine from Lemma 13.

The total time complexity of solving the problem exactly over d'_ℓ equals the running time of the subroutine times the number of possible configurations, which is $\left(\mathcal{O}\left(\ell \cdot \frac{1}{\varepsilon} \ln \frac{n}{\varepsilon}\right)\right)^k n^{\mathcal{O}(1)}$.

It remains to prove that the algorithm yields a proper approximation. We will show that for any solution SOL it holds that

$$cost\left(\phi^{SOL}, d_\ell\right) \leqslant cost\left(\phi^{SOL}, d'_\ell\right) \leqslant (1+\varepsilon) \cdot cost\left(\phi^{SOL}, d_\ell\right) + \varepsilon \cdot D. \tag{1}$$

By substituting $SOL = OPT(d_\ell)$ we learn that there exists a solution over metric d'_ℓ of cost at most $(1+\varepsilon) \cdot cost\left(\phi^{OPT(d_\ell)}, d_\ell\right) + \varepsilon \cdot D \leqslant (1+2\varepsilon) \cdot cost\left(\phi^{OPT(d_\ell)}, d_\ell\right)$ for correctly guessed D. Therefore the cost of the solution found by our algorithm cannot be larger. Finally we substitute this solution as SOL to see that its cost cannot increase when returning to metric d_ℓ. The claim will follow by adjusting ε.

The first inequality in (1) is straightforward because d'_ℓ dominates d_ℓ. Consider now a pair $(c, f = \phi^{SOL}(c))$, where $f \in F_i(s)$. If $i < \lceil \log_{1+\varepsilon} \frac{n}{\varepsilon} \rceil$, then $d_\ell(c,f) \leqslant d'_\ell(c,f) \leqslant (1+\varepsilon) \cdot d_\ell(c,f)$, so the cost of connecting such pairs increases at most by a multiplicative factor $(1+\varepsilon)$ during the metric switch. If $i = \lceil \log_{1+\varepsilon} \frac{n}{\varepsilon} \rceil$, then $d'_\ell(s,f) = \frac{\varepsilon D}{n}$. Since there are at most n such pairs, the total additive cost increase is bounded by $\varepsilon \cdot D$. ◀

▶ **Theorem 16.** Non-Uniform CKM *can be solved with approximation ratio* $(7+\epsilon)$ *in time* $(k/\epsilon)^{\mathcal{O}(k)} n^{\mathcal{O}(1)}$.

Proof. From Lemma 15 we know that we can get a $(1+\varepsilon)$-approximation algorithm for the Non-Uniform CKM problem on ℓ-centered instances in time $\left(\mathcal{O}\left(\ell \cdot \frac{1}{\varepsilon} \ln \frac{n}{\varepsilon}\right)\right)^k n^{\mathcal{O}(1)}$. We shall use the $(1+\varepsilon)$-approximation for Uncapacitated KM by Lin and Vitter [24], that opens at most $\ell = \left(1 + \frac{1}{\varepsilon}\right) k \cdot (\ln n + 1)$ facilities. By plugging this subroutine to find ℓ-centers into the Lemma 7 together with Lemma 15, we obtain a $(7+\varepsilon)$-approximation algorithm for the general Non-Uniform CKM problem with running time

$$\mathcal{O}\left(\left(\left(1+\frac{1}{\varepsilon}\right) k \cdot (\ln n + 1) \cdot \frac{1}{\varepsilon} \ln \frac{n}{\varepsilon}\right)^k\right) n^{\mathcal{O}(1)} = \mathcal{O}\left(\left(\frac{1}{\varepsilon^{\mathcal{O}(1)}} k \ln^2 n\right)^k\right) n^{\mathcal{O}(1)}.$$

Finally, we use standard arguments to show that $(\ln n)^{2k} \leqslant \max(n, k^{\mathcal{O}(k)})$. Consider two cases. If $\frac{\ln n}{2 \ln \ln n} \leqslant k$, then by inverting we know that $\ln n = \mathcal{O}(k \ln k)$, and so $(\ln n)^{2k} = k^{\mathcal{O}(k)}$. Suppose now that $\frac{\ln n}{2 \ln \ln n} > k$. In this case

$$(\ln n)^{2k} < (\ln n)^{\frac{\ln n}{\ln \ln n}} = e^{\ln \ln n \cdot \frac{\ln n}{\ln \ln n}} = n. \quad ◀$$

5 Conclusions and open problems

We have presented a $(7+\varepsilon)$-approximation algorithm for the CKM problem, which consists of three building blocks: approximation for Uncapacitated KM, metric embedding into a simpler structure, and a parameterized algorithm working on ℓ-centered instances.

Whereas the first and the last ingredient are almost lossless from the approximation point of view, the embedding procedure seems to be the main bottleneck for obtaining a better approximation guarantee. One can imagine that a different technique would allow to obtain a $(1+\varepsilon)$-approximation in FPT time. We believe that finding such an algorithm or ruling out its existence is an interesting research direction.

Another avenue for improvement is processing k-centered instances in time $2^{\mathcal{O}(k)} n^{\mathcal{O}(1)}$. Such a routine would reduce the running time of the whole algorithm to single exponential. In order to do so, one could replace the subroutine for Uncapacitated KM by Lin and Vitter [24] with a standard approximation algorithm that opens exactly k facilities, what would moderately increase the constant in approximation ratio.

Finally, whereas we have used the framework of ℓ-centered instances to devise an FPT approximation, it might be possible to explore the structure of special instances further and find a polynomial time approximation algorithm. This could yield an improvement over the $\mathcal{O}(\log k)$-approximation ratio for CKM, which remains a major open problem.

References

1. Karen Aardal, Pieter L van den Berg, Dion Gijswijt, and Shanfei Li. Approximation algorithms for hard capacitated k-facility location problems. *European Journal of Operational Research*, 242(2):358–368, 2015.
2. Vijay Arya, Naveen Garg, Rohit Khandekar, Adam Meyerson, Kamesh Munagala, and Vinayaka Pandit. Local search heuristics for k-median and facility location problems. *SIAM Journal on Computing*, 33(3):544–562, 2004.
3. Jarosław Byrka, Krzysztof Fleszar, Bartosz Rybicki, and Joachim Spoerhase. Bi-factor approximation algorithms for hard capacitated k-median problems. In *Proceedings of the Twenty-Sixth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 722–736. SIAM, 2015.

4 Jarosław Byrka, Thomas Pensyl, Bartosz Rybicki, Aravind Srinivasan, and Khoa Trinh. An improved approximation for k-median, and positive correlation in budgeted optimization. In *Proceedings of the Twenty-Sixth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 737–756. SIAM, 2015.

5 Jaroslaw Byrka, Bartosz Rybicki, and Sumedha Uniyal. An Approximation Algorithm for Uniform Capacitated k-Median Problem with $1+\epsilon$ Capacity Violation. In *Integer Programming and Combinatorial Optimization - 18th International Conference, IPCO 2016, Liège, Belgium, June 1-3, 2016, Proceedings*, pages 262–274, 2016. doi:10.1007/978-3-319-33461-5_22.

6 Parinya Chalermsook, Marek Cygan, Guy Kortsarz, Bundit Laekhanukit, Pasin Manurangsi, Danupon Nanongkai, and Luca Trevisan. From gap-ETH to FPT-inapproximability: Clique, dominating set, and more. In *Foundations of Computer Science (FOCS), 2017 IEEE 58th Annual Symposium on*, pages 743–754. IEEE, 2017.

7 Moses Charikar, Chandra Chekuri, Ashish Goel, and Sudipto Guha. Rounding via Trees: Deterministic Approximation Algorithms for Group Steiner Trees and k-Median. In *Proceedings of the Thirtieth Annual ACM Symposium on the Theory of Computing, Dallas, Texas, USA, May 23-26, 1998*, pages 114–123, 1998. doi:10.1145/276698.276719.

8 Moses Charikar, Chandra Chekuri, Ashish Goel, Sudipto Guha, and Serge A. Plotkin. Approximating a Finite Metric by a Small Number of Tree Metrics. In *39th Annual Symposium on Foundations of Computer Science, FOCS '98, November 8-11, 1998, Palo Alto, California, USA*, pages 379–388, 1998. doi:10.1109/SFCS.1998.743488.

9 Moses Charikar, Sudipto Guha, Éva Tardos, and David B Shmoys. A constant-factor approximation algorithm for the k-median problem. In *Proceedings of the thirty-first annual ACM symposium on Theory of computing*, pages 1–10. ACM, 1999.

10 Julia Chuzhoy and Yuval Rabani. Approximating k-median with non-uniform capacities. In *Proceedings of the sixteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 952–958. Society for Industrial and Applied Mathematics, 2005.

11 Vincent Cohen-Addad and Jason Li. On the Fixed-Parameter Tractability of Capacitated Clustering. In *to appear in the 46th International Colloquium on Automata, Languages and Programming (ICALP)*, 2019.

12 Marek Cygan, MohammadTaghi Hajiaghayi, and Samir Khuller. LP rounding for k-centers with non-uniform hard capacities. In *Foundations of Computer Science (FOCS), 2012 IEEE 53rd Annual Symposium on*, pages 273–282. IEEE, 2012.

13 Artur Czumaj and Christian Sohler. Small space representations for metric min-sum k-clustering and their applications. *Theory of Computing Systems*, 46(3):416–442, 2010.

14 H. Gökalp Demirci and Shi Li. Constant Approximation for Capacitated k-Median with $(1+\epsilon)$-Capacity Violation. In *43rd International Colloquium on Automata, Languages, and Programming, ICALP 2016, July 11-15, 2016, Rome, Italy*, pages 73:1–73:14, 2016. doi:10.4230/LIPIcs.ICALP.2016.73.

15 Jittat Fakcharoenphol, Satish Rao, and Kunal Talwar. A tight bound on approximating arbitrary metrics by tree metrics. In *Proceedings of the 35th Annual ACM Symposium on Theory of Computing, June 9-11, 2003, San Diego, CA, USA*, pages 448–455, 2003. doi:10.1145/780542.780608.

16 Fedor V. Fomin, Daniel Lokshtanov, Neeldhara Misra, and Saket Saurabh. Planar F-Deletion: Approximation, Kernelization and Optimal FPT Algorithms. In *Proceedings of the 2012 IEEE 53rd Annual Symposium on Foundations of Computer Science*, FOCS '12, pages 470–479, Washington, DC, USA, 2012. IEEE Computer Society.

17 Anupam Gupta, Euiwoong Lee, and Jason Li. An FPT algorithm beating 2-approximation for k-cut. In *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 2821–2837. Society for Industrial and Applied Mathematics, 2018.

18 Anupam Gupta, Euiwoong Lee, Jason Li, Pasin Manurangsi, and Michal Wlodarczyk. Losing Treewidth by Separating Subsets. *CoRR*, abs/1804.01366, 2018. arXiv:1804.01366.

19 Madhukar R Korupolu, C Greg Plaxton, and Rajmohan Rajaraman. Analysis of a local search heuristic for facility location problems. *Journal of algorithms*, 37(1):146–188, 2000.

20 Euiwoong Lee. Partitioning a graph into small pieces with applications to path transversal. *Mathematical Programming*, 2018. Preliminary version in SODA 2017.

21 Shi Li. On uniform capacitated k-median beyond the natural LP relaxation. In *Proceedings of the Twenty-Sixth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 696–707. SIAM, 2015.

22 Shi Li. Approximating capacitated k-median with $(1+\epsilon)k$ open facilities. In *Proceedings of the Twenty-Seventh Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), To Appear*, 2016.

23 Shi Li and Ola Svensson. Approximating k-median via pseudo-approximation. In *proceedings of the forty-fifth annual ACM symposium on theory of computing*, pages 901–910. ACM, 2013.

24 Jyh-Han Lin and Jeffrey Scott Vitter. ε-Approximations with Minimum Packing Constraint Violation (Extended Abstract). In *Proceedings of the Twenty-fourth Annual ACM Symposium on Theory of Computing*, STOC '92, pages 771–782, New York, NY, USA, 1992. ACM. doi:10.1145/129712.129787.

25 Karthik C. S., Bundit Laekhanukit, and Pasin Manurangsi. On the Parameterized Complexity of Approximating Dominating Set. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*, STOC 2018, pages 1283–1296, New York, NY, USA, 2018. ACM. doi:10.1145/3188745.3188896.

26 Alexander Schrijver. *Combinatorial Optimization – Polyhedra and Efficiency*. Springer, 2003.

27 Yicheng Xu, Yong Zhang, and Yifei Zou. A constant parameterized approximation for hard-capacitated k-means. *CoRR*, abs/1901.04628, 2019. arXiv:1901.04628.