Constant Factor Approximation for Capacitated k-Center with Outliers*

Marek Cygan and Tomasz Kociumaka

Institute of Informatics, University of Warsaw, Poland
{cygan, kociumaka}@mimuw.edu.pl

Abstract

The k-center problem is a classic facility location problem, where given an edge-weighted graph $G = (V, E)$ one is to find a subset of k vertices S, such that each vertex in V is "close" to some vertex in S. The approximation status of this basic problem is well understood, as a simple 2-approximation algorithm is known to be tight. Consequently different extensions were studied.

In the capacitated version of the problem each vertex is assigned a capacity, which is a strict upper bound on the number of clients a facility can serve, when located at this vertex. A constant factor approximation for the capacitated k-center was obtained last year by Cygan, Hajiaghayi and Khuller [FOCS'12], which was recently improved to a 9-approximation by An, Bhaskara and Svensson [arXiv'13].

In a different generalization of the problem some clients (denoted as outliers) may be disregarded. Here we are additionally given an integer p and the goal is to serve exactly p clients, which the algorithm is free to choose. In 2001 Charikar et al. [SODA'01] presented a 3-approximation for the k-center problem with outliers.

In this paper we consider a common generalization of the two extensions previously studied separately, i.e. we work with the capacitated k-center with outliers. We present the first constant factor approximation algorithm with approximation ratio of 25 even for the case of non-uniform hard capacities.

1998 ACM Subject Classification F.2.2 Nonnumerical Algorithms and Problems

Keywords and phrases approximation algorithms, k-center, capacities, outliers, LP rounding

Digital Object Identifier 10.4230/LIPIcs.STACS.2014.251

1 Introduction

The k-center problem is a classic facility location problem and is defined as follows: given a finite set V and a symmetric distance (cost) function $d : V \times V \to \mathbb{R}_{\geq 0}$ satisfying the triangle inequality, find a subset $S \subseteq V$ of size k such that each vertex in V is "close" to some vertex in S. More formally, once we choose S the objective function to be minimized is $\max_{v \in V} \min_{u \in S} d(v, u)$. The vertices of S are called *centers* or *facilities*. The problem is known to be NP-hard [12]. Approximation algorithms for the k-center problem have been well studied and are known to be optimal [13, 15, 16, 17].

In the capacitated setting, studied for twenty years already, we are additionally given a capacity function $L : V \to \mathbb{Z}_{\geq 0}$ and no more than $L(u)$ vertices (called *clients*) may be assigned to a chosen center at $u \in V$. For the special case when all the capacities are *identical* (denoted as the *uniform* case), a 6-approximation was developed by Khuller and Sussmann [19] improving the previous bound of 10 by Bar-Ilan, Kortsarz and Peleg [4]. In the

* This work is partially supported by Foundation for Polish Science grant HOMING PLUS/2012-6/2.

31st Symposium on Theoretical Aspects of Computer Science (STACS'14).
Editors: Ernst W. Mayr and Natacha Portier; pp. 251–262
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

soft capacities version, in contrast to the standard (*hard* capacities), we are allowed to open several facilities in a single location, i.e. the facilities may form a multiset. For the uniform soft capacities version the best known approximation ratio equals 5 [19]. For general hard capacities a constant factor approximation has been obtained only recently [11], somewhat surprisingly by using LP rounding. It was followed by a cleaner and simpler approach of An, Bhaskara and Svensson [1] who gave a 9-approximation algorithm. From the hardness perspective a $(3 - \varepsilon)$ lower bound on the approximation ratio is known [9, 11].

Another natural direction in generalizing the problem is an assumption that instead of serving all the clients we are given an integer p and we are to select exactly p clients to serve. The disregarded clients are in the literature called *outliers*. The k-center problem with outliers admits a 3-approximation algorithm, which was obtained by Charikar et al. [8].

In this article we study a common generalization of the two mentioned variants of the k-center problem, i.e. involving both capacities and outliers. In order to simplify our algorithms we work with a slight generalization, the Capacitated k-supplier with Outliers problem, where vertices are either clients or potential facility locations. These vertices may coincide, so that one may have both a client and a potential facility location at the same point, as in k-center. Below we give the formal problem definition.

Capacitated k-supplier with Outliers
Input: Integers $k, p \in \mathbb{Z}_{\geq 0}$, finite sets $\mathcal{C}$ and $\mathcal{F}$, a symmetric distance (cost) function $d : (\mathcal{C} \cup \mathcal{F}) \times (\mathcal{C} \cup \mathcal{F}) \to \mathbb{R}_{\geq 0}$ satisfying the triangle inequality, and a capacity function $L : \mathcal{F} \to \mathbb{Z}_{\geq 0}$
Find: Sets $C \subseteq \mathcal{C}$, $F \subseteq \mathcal{F}$, and a function $\phi : C \to F$ satisfying

- $|C| = p$,
- $|F| = k$,
- $|\phi^{-1}(u)| \leq L(u)$ for each $u \in F$.

Minimize: $\max_{v \in C} d(v, \phi(v))$.

Again, in the *soft* capacities version, F is allowed to be a multiset, and in the *uniform* capacities version, the capacity function L is constant.

Existence of an r-approximation algorithm for Capacitated k-center with Outliers can be shown to be equivalent to existence of an r-approximation algorithm for Capacitated k-supplier with Outliers.[1] Interestingly, such an equivalence is not known to hold if we do not allow outliers: the best known approximation factor for the Capacitated k-supplier is 11 while for the Capacitated k-center it is 9, see [1].

1.1 Our results and organization of the paper

The following is the main result of this paper.

▶ **Theorem 1.** *The* Capacitated k-supplier with Outliers *problem, both in hard and soft capacities version, admits a 25-approximation algorithm. The hard uniform capacities version admits a 23-approximation, and soft uniform capacities – a 13-approximation.*

Note that taking $\mathcal{C} = \mathcal{F} = V$ shows that the k-supplier problem generalizes the k-center problem, and consequently gives the same approximation bounds for the latter.

▶ **Corollary 2.** *The* Capacitated k-center with Outliers *problem, both in hard and soft capacities version, admits a 25-approximation algorithm. The hard uniform capacities version admits a 23-approximation, and soft uniform capacities – a 13-approximation.*

[1] The proof is left for the full version of the paper.

It is worth noting, that the already known approximation algorithm for the k-center problem with outliers relies on the fact that a single vertex can serve all the clients that are its neighbors, i.e. there are no capacity constraints. At the same time the previous approximation algorithms for the capacitated k-center problem (both in the uniform and non-uniform case) heavily used the fact that each vertex of the graph is close to some center in any solution. For this reason it was possible to create a path-like [11] or tree-like [1] structure with integrally opened non-leaf vertices, that was the crux in the rounding process. Consequently none of the algorithms for the two previously independently studied extensions of the basic problem, i.e. capacities and outliers, works for the problem we are interested in.

The first step of our algorithm (Section 3) is the standard thresholding technique, where we reduce a general metric to a distance metric of an unweighted graph. In Section 4 we introduce our main conceptual contribution, i.e. the notion of a *skeleton*. A skeleton is a set S of vertices, for which there exists an optimum solution $F \subseteq \mathcal{F}$, such that each vertex of S can be injectively mapped to a nearby vertex of F and moreover each vertex of F is close to some vertex of S. Intuitively a skeleton is not yet a solution, but it looks similar to at least one optimum solution. If no outliers are allowed, any inclusionwise maximal subset of $\mathcal{F}$ with vertices far enough from each other, is a skeleton. In [11] and [1], such a set is then mapped to non-leaf vertices of the structure steering the rounding process. We use a skeleton in a similar way, but before we are able to do that, we need to bound the integrality gap. Without outliers, it was sufficient to take the standard LP relaxation and decompose the graph into connected components. Although with outliers this is no longer the case, as shown in Section 5, a skeleton lets us both strengthen the LP relaxation, adding an appropriate constraint, and obtain a more granular decomposition of the initial instance into several subinstances, for which the strengthened LP relaxation is feasible and has bounded integrality gap. Further in Section 6 we show how each of these smaller instances can be independently rounded using tools previously applied for the capacitated setting [1].[2] Section 7 contains a wrap-up of the whole algorithm.

The improvements in the approximation ratio when soft or uniform capacities are considered, are postponed to the full version of the paper.

1.2 Related facility location work

The *facility location* problem is a central problem in operations research and computer science and has been a testbed for many new algorithmic ideas resulting a number of different approximation algorithms. In this problem, given a metric (via a weighted graph G), a set of nodes called *clients*, and opening costs on some nodes called *facilities*, the goal is to open a subset of facilities such that the sum of their opening costs and connection costs of clients to their nearest open facilities is minimized. Up to now, the best known approximation ratio is 1.488, due to Li [21] who used a randomized selection in Byrka's algorithm [6]. Guha and Khuller [14] showed that this problem is hard to approximate within a factor better than 1.463, assuming $NP \not\subseteq DTIME[n^{O(\log\log n)}]$.

When the facilities have capacities, the problem is called the *capacitated facility location* problem. It has also received a great deal of attention in recent years. Two main variants of the problem are soft-capacitated facility location and hard-capacitated facility location: in the latter problem, each facility is either opened at some location or not, whereas in

[2] The final rounding step can be also done using the path-like structures notion of [11], however we use the ideas of [1] as it allows cleaner presentation.

the former, one may specify any integer number of facilities to be opened at that location. Soft capacities make the problem easier and by modifying approximation algorithms for the uncapacitated problems, we can also handle this case [23, 18]. To the best of our knowledge all the existing constant-factor approximation algorithms for the general case of hard capacitated facility location are local search based, and the most recent of them is the 5-approximation algorithm of Bansal, Garg and Gupta [3]. The only LP-relaxation based approach for this problem is due to Levi, Shmoys and Swamy [20] who gave a 5-approximation algorithm for the special case in which all facility opening costs are equal (otherwise the LP does not have a constant integrality gap). Obtaining an LP based constant factor approximation algorithm for capacitated facility location is considered a major problem in approximation algorithms [24].

A problem very close to both facility location and k-center is the *k-median* problem in which we want to open at most k facilities and the goal is to minimize the sum of connection costs of clients to their nearest open facilities. Very recently Li and Svensson [22] obtained an LP rounding $(1+\sqrt{3})$-approximation algorithm, improving upon the previously best $(3+\varepsilon)$-approximation local search algorithm of Arya et al. [2]. Unfortunately obtaining a constant factor approximation algorithm for capacitated k-median still remains open despite consistent effort. The only previous attempts with constant approximation factors for this problem violate the capacities within a constant factor for the uniform capacity case [7] and the non-uniform capacity case [10] or exceed the number k of facilities by a constant factor [5].

2 Preliminaries

For a fixed instance of the Capacitated k-supplier with Outliers, we call (C, F, ϕ) *a solution* if it satisfies the required conditions. We often identify the solution by ϕ only (considering it as a partial function from $\mathcal{C}$ to $\mathcal{F}$), using C_ϕ and F_ϕ to refer to the other elements of the triple. If ϕ satisfies $\max_{v \in C} d(v, \phi(v)) \leq \tau$, we say that ϕ is a distance-τ solution.

Let $G = (V, E)$ be an undirected graph. By d_G we denote the metric defined by G. For sets $A, B \subseteq V$ we define $d_G(A, B) = \min_{a \in A, b \in B} d_G(a, b)$. If $B = \{b\}$ we write $d_G(A, b)$ instead of $d_G(A, B)$.

For a vertex $v \in V$ and an integer $k \in \mathbb{Z}_{\geq 0}$ we denote $N_G^k(v) = \{u \in V : d_G(u, v) = k\}$ and $N_G^k[v] = \{u \in V : d_G(u, v) \leq k\}$. We omit the superscript for $k = 1$ and the subscript if there is no confusion which graph we refer to.

For a set S and an element s by $S + s$ we denote $S \cup \{s\}$.

3 Reduction to graphic instances

As usual when working with a min max problem we start with the standard thresholding argument, i.e. reduce a general metric function to a metric defined by an unweighted graph.

We say that an instance of the k-supplier problem is *graphic*, if d is defined as the distance function of an unweighted bipartite graph $G = (\mathcal{C}, \mathcal{F}, E)$, and the goal is to find a distance-1 solution. An r-approximation algorithm is then allowed to either give a distance-r solution, or, only if it finds out that no distance-1 solution exists, a NO answer.

Below we show how to build an r-approximation algorithm for Capacitated k-supplier with Outliers given an r-approximation (in the aforementioned sense) for the graphic instances. Correctness of the reduction is standard. If an optimal solution exists, then its value OPT belongs to T. In particular, in the phase corresponding to OPT, there is a distance-1 solution in $G_{\leq OPT}$. Thus the algorithm for graphic instances is required to find a solution. Therefore returns a solution ϕ for the first time at phase corresponding to $\tau^* \leq OPT$. Since $d(v,u) \leq \tau^* d_{G_{\leq \tau^*}}(v,u)$, ϕ is a distance-$r \cdot \tau^*$ solution, hence also distance-$r \cdot OPT$ solution.

```
$T := \{d(v,u) : v \in \mathcal{C}, u \in \mathcal{F}\}$;
foreach $\tau \in T$ in ascending order do
    $G_{\leq\tau} := (\mathcal{C}, \mathcal{F}, \{(v,u) : d(v,u) \leq \tau\})$;
    solve the graphic instance for $G_{\leq\tau}$;
    if a solution $\phi$ found then return $\phi$;
return $NO$;
```

Algorithm 1: Reduction to graphic instances

4 Finding a skeleton

From now on we work with graphic instances only. Without loss of generality we may assume that $L(u) \leq deg(u)$ for each $u \in \mathcal{F}$. Indeed, setting $L(u) := \min(L(u), deg(u))$ has no influence on distance-1 solutions, while no additional distance-r solutions are created.

The first phase of the algorithm outputs several subsets of $\mathcal{F}$. If a distance-1 solution exists, at least one of them resembles (in a certain sense, to be defined later) a distance-1 solution and can be successfully used by the subsequent phases as a hint for constructing a distance-r solution. We formalize the features of a good hint in the following definition.

▶ **Definition 3.** A set $S \subseteq \mathcal{F}$ is called a *skeleton* if

- (**separation property**) $d(u,u') \geq 6$ for any $u, u' \in S$, $u \neq u'$,
- there exists a distance-1 solution (C_ϕ, F_ϕ, ϕ) such that:
  - (**covering property**) $d(u,S) \leq 4$ for each $u \in F_\phi$,
  - (**injection property**) there exists an injection $f : S \hookrightarrow F_\phi$ satisfying $d(u, f(u)) \leq 2$ for each $u \in S$.

If just separation and injection properties are satisfied, we call S a *preskeleton*.

In other words a skeleton is a set S, each vertex of which can be injectively mapped to a vertex of a distance-1 solution F_ϕ, and at the same time no two vertices of S are close and $N^4[S]$ contains the whole set F_ϕ.

Note that the separation property implies that sets $N^2[u]$ are pairwise disjoint for $u \in S$, hence any function $f : S \to F_\phi$ satisfying $d(u, f(u)) \leq 2$ is in fact an injection, however we make it explicit for the sake of presentation.

▶ **Lemma 4.** *Let S be a preskeleton and let $U = \{u \in \mathcal{F} : d(u,S) \geq 6\}$. Then S is a skeleton, or $U \neq \emptyset$ and $S + s$ is a preskeleton, where s is a highest-capacity vertex of U.*

Proof. Let ϕ be a distance-1 solution, which witnesses S being a preskeleton, where $f : S \hookrightarrow F_\phi$ satisfies the injection property. If ϕ witnesses S being a skeleton, we are done. Otherwise the covering property is not satisfied, hence there exists $u \in F_\phi$ such that $d(u,S) > 4$. Since d is a distance function of a bipartite graph, this implies $d(u,S) \geq 6$, so $u \in U \neq \emptyset$. If $|F_\phi \cap N^2[s]| \geq 1$, then ϕ already witnesses $S + s$ being a preskeleton, as one can extend the injection f by mapping a vertex of $F_\phi \cap N^2[s]$ to s. Therefore, we may assume that $N^2[s] \cap F_\phi = \emptyset$. In particular, this means that the clients in $N(s)$ are not served by any facility of F_ϕ.

Let us modify ϕ to obtain ψ as follows: close the facility in u, opening one in s instead. Let c be the number of clients assigned to u in ϕ. No longer serve these, instead serve any c neighbors of s in ψ (as we have observed before, they are not served in ϕ). Note that $c \leq L(u) \leq L(s) \leq \deg(s)$ by the choice of u maximizing the capacity and by the assumption of L being bounded by deg. Consequently, there are enough neighbors of s to serve, and the capacity constraint for s is satisfied. Moreover, the number of open facilities and the number of served clients are preserved. Other open facilities remain unchanged, so ψ satisfies the capacity and distance constraints for them, and therefore is a distance-1 solution. Finally, consider a function $f' = f + (s, s)$. As s is at distance at least 6 from S, by the injection property for S we know that s does not belong to the image of f, hence f' is an injection. Consequently ψ and f' ensure $S + s$ satisfies the injection property. Moreover s is far from S, hence $S + s$ is a preskeleton. ◀

With $\emptyset$ being trivially a preskeleton provided that any distance-1 solution exists, Lemma 4 lets us generate a sequence of sets, which contains a skeleton (see Algorithm 2). Note that any skeleton, by the injection property, is of size at most k.

▶ **Lemma 5.** *If there exists a distance-1 solution, there is at least one skeleton among sets output by Algorithm 2.*

```
$S := \emptyset$;
while $|S| \leq k - 1$ do
    $U := \{u \in \mathcal{F} \;:\; d(u, S) \geq 6\}$;
    if $U = \emptyset$ then break;
    $s := \operatorname{argmax}\{L(u) \;:\; u \in U\}$;
    $S := S + s$;
    output $S$;
```

Algorithm 2: Construction of a family of sets containing at least one skeleton.

5 Clustering

For a set $S \subseteq \mathcal{F}$ define the following linear program $LP_{k,p}(G, L, S)$, where a variable y_u for $u \in \mathcal{F}$ denotes whether we open a facility in u or not, while a variable x_{uv} for $u \in \mathcal{F}$, $v \in \mathcal{C}$ corresponds to whether u serves v or not.

$$\sum_{u \in \mathcal{F}} y_u = k \tag{1}$$

$$\sum_{u \in \mathcal{F}, v \in \mathcal{C}} x_{uv} = p \tag{2}$$

$$x_{uv} \leq y_u \qquad \text{for each } u \in \mathcal{F}, v \in \mathcal{C} \tag{3}$$

$$\sum_{v} x_{uv} \leq L(u) \cdot y_u \qquad \text{for each } u \in \mathcal{F} \tag{4}$$

$$\sum_{u} x_{uv} \leq 1 \qquad \text{for each } v \in \mathcal{C} \tag{5}$$

$$\sum_{u \in \mathcal{F} \cap N^2[s]} y_u \geq 1 \qquad \text{for each } s \in S \tag{6}$$

$$x_{uv} = 0 \qquad \text{for each } u \in \mathcal{F}, v \in \mathcal{C} \text{ such that } (v, u) \notin E \tag{7}$$

$$\mathbf{0} \leq x, y \leq \mathbf{1} \tag{8}$$

Constraints (1)–(5),(7) are the standard constraints for Capacitated k-supplier with Outliers, ensuring that we open exactly k facilities (1), serve exactly p clients (2), obey capacity constraints (3)–(5), and serve clients which are close to facilities (7).

Observe that if S is a skeleton and a distance-1 solution ϕ witnesses that fact, we get a feasible solution of $LP_{k,p}(G, L, S)$ setting $y_u = 1$ iff $u \in F_\phi$ and $x_{uv} = 1$ iff $v \in C_\phi$ and $v = \phi(u)$. Indeed the injection property ensures that constraint (6) is satisfied. However, as usual in a capacitated problem with hard constraints, the integrality gap of this LP is unbounded. Similarly to the standard capacitated k-center [11], this issue is addressed by considering the connected components of G separately. When all the clients need to be served having a connected graph with a feasible solution of the standard LP is enough to round it [1, 11]. However, if we allow outliers, there are sill connected instances with arbitrarily large integrality gap.[3] For this reason we use the additional constraint (6) together with the assumption that all the vertices are close to S. This way we crucially exploit the covering, injection and separation properties of a skeleton.

In the following we shall prove that any instance with a skeleton can be decomposed into several smaller instances with additional properties. In the next section we will show how to round the obtained smaller instances.

▶ **Lemma 6.** *Let $S \subseteq \mathcal{F}$, let $G_1, \dots, G_\ell$ be components of G after all vertices v with $d(v, S) > 5$ are removed and let $S_i = S \cap V(G_i)$ for $1 \le i \le \ell$.*

If S is a skeleton, then in polynomial time one can find partitions $k = \sum_{i=1}^{\ell} k_i$ and $p = \sum_{i=1}^{\ell} p_i$ such that $LP_{k_i,p_i}(G_i, L, S_i)$ are all feasible.

Proof. Observe that if S is a skeleton, then a witness solution ϕ opens facilities at distance at most 4 from S, and thus serves clients with distance at most 5 from S. Consequently all vertices further from S can be safely removed and S remains a skeleton. Then G might contain several connected components $G_1, \dots, G_\ell$ with $G_i = (\mathcal{C}_i, \mathcal{F}_i, E_i)$. The witness solution ϕ can be partitioned among these components so that we get assignments ϕ_i which in total open k facilities to serve p clients. In particular, this means that for some partitions $k = \sum_i k_i$ and $p = \sum_i p_i$ sets $S_i = S \cap \mathcal{F}_i$ are skeletons, and consequently $LP_{k_i,p_i}(G_i, L, S_i)$ are feasible. The latter condition can be tested efficiently for any values k_i and p_i. While we cannot exhaustively test all partitions of k and p, dynamic programming lets us find partitions such that these linear programs are feasible for each i.

For $i \in \{0, \dots, \ell\}$, $k' \in \{0, \dots, k\}$ and $p' \in \{0, \dots, p\}$ define a boolean value $F[i][k'][p']$, which equals **true** iff there exist partitions $k' = \sum_{j=1}^{i} k_j$ and $p' = \sum_{j=1}^{i} p_j$ such that $LP_{k_j,p_j}(G_j, L, S_j)$ are all feasible for $j \le i$.

Clearly $F[0][0][0]$ is true, while $F[0][k'][p']$ is false for any other pair (k', p'). For $i > 1$ the value $F[i][k'][p']$ is simply an alternative of $F[i-1][k'-k_i][p'-p_i]$ for every pair (k_i, p_i) such that $LP_{k_i,p_i}(G_i, L, S_i)$ is feasible, $k_i \le k'$ and $p_i \le p'$. Thus in polynomial time one can check whether the desired partitions exists, and provided that together with a true value we also store the witness partitions, also find these partitions. ◀

6 Rounding

In the previous section we have shown how given a skeleton S one can partition the initial instance into smaller subinstances with more structural properties. Our main goal in this

[3] The construction is simple, but due to space restrictions it is left for the full version of the paper.

section is to show that those structural properties are in fact sufficient to construct a solution for each of the subinstances, which is formalized in the following lemma.

▶ **Lemma 7.** *Let $I = (G = (\mathcal{C}, \mathcal{F}, E), L, k, p)$ be an instance of* Capacitated k-supplier with Outliers *and let $S \subseteq \mathcal{F}$. If the following four conditions are satisfied:*

(i) *G is connected,*
(ii) *for any $u, u' \in S$, $u \neq u'$ we have $d(u, u') \geq 6$,*
(iii) *$N^5[S] = \mathcal{F} \cup \mathcal{C}$,*
(iv) *$LP_{k,p}(G, L, S)$ admits a feasible solution,*

then one can find a distance-25 solution for I in polynomial time.

Before we give a proof of Lemma 7, in Section 6.1 we recall (an adjusted version) of a distance-r transfer, a very useful notion introduced in [1], together with its main properties. Next, in Section 6.2 we prove Lemma 7.

6.1 Distance r-transfer

▶ **Definition 8.** Given a graph $G = (V, E)$ with $W \subseteq V$, a capacity function $L : W \to \mathbb{Z}_{\geq 0}$ and $y \in \mathbb{R}^W_{\geq 0}$, a vector $y' \in \mathbb{R}^W_{\geq 0}$ is a distance-r transfer of (G, L, y) if

1. $\sum_{v \in W} y'_v = \sum_{v \in W} y_v$ and
2. $\sum_{v \in W : d(v,U) \leq r} L(v) y'_v \geq \sum_{u \in U} L(u) y_u$ for all $U \subseteq W$.

If y' is a characteristic vector of $F \subseteq W$, we say that F is an integral distance-r transfer of (G, L, y).

Less formally a distance-r transfer is a reassignment, where the sum of y-variables is preserved and locally for any set $U \subseteq W$ the total fractional capacity in a small neighborhood of U does not decrease.

Like in [1], an integral distance-r transfer of the fractional solution of the LP already gives a distance-$r + 1$ solution (in particular point 2 of Definition 8 ensures that the Hall's condition is satisfied). The proof must be modified though, so that it encompasses outliers.

▶ **Lemma 9.** *Let $G = (\mathcal{C}, \mathcal{F}, E)$ be a bipartite graph with a capacity function $L : \mathcal{F} \to \mathbb{Z}_{\geq 0}$. Assume (x, y) is a feasible solution of $LP_{k,p}(G, L, S)$ and $F \subseteq \mathcal{F}$ is an integral distance-r transfer of y. Then one can find a distance-$r + 1$ solution (C, F, ϕ) in polynomial time.*

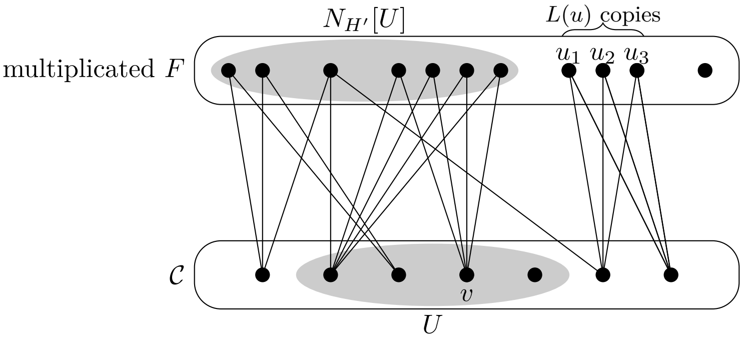

Figure 1 Graph H' obtained from H by removing vertices from $\mathcal{F} \setminus F$ and duplicating each vertex $u \in F$ to its capacity. Shaded ellipses represent sets used in Hall's theorem.

Proof. Consider a bipartite graph $H = (\mathcal{C}, \mathcal{F}, E_H)$ with $(v,u) \in E_H$ if $d_G(v,u) \le r+1$. Modify H to obtain H' by removing vertices from $\mathcal{F} \setminus F$ and duplicating each vertex $u \in F$ to its capacity, i.e. $L(u)$ times, see also Fig. 1. Observe that cardinality-p matchings in this graph correspond to distance-$r+1$ solutions for G. If any, such a matching can clearly be found in polynomial time. We shall prove its existence by checking the deficit version of Hall's theorem, i.e. that for each $U \subseteq \mathcal{C}$ we have

$$\sum_{u \in F : d(u,U) \le r+1} L(u) \ge |U| - |\mathcal{C}| + p$$

First, observe that

$$\sum_{v \in U, u \in \mathcal{F}} x_{uv} = \sum_{v \in \mathcal{C}, u \in \mathcal{F}} x_{uv} - \sum_{v \in \mathcal{C} \setminus U, u \in F} x_{uv} \overset{(2),(5)}{\ge} p - \sum_{v \in \mathcal{C} \setminus U} 1 = p - |\mathcal{C} \setminus U| = |U| - |\mathcal{C}| + p.$$

Moreover

$$\sum_{v \in U, u \in \mathcal{F}} x_{uv} = \sum_{v \in U, u \in N_G(U)} x_{uv} \le \sum_{u \in N_G(U)} \sum_{v \in \mathcal{C}} x_{uv} \overset{(4)}{\le} \sum_{u \in N_G(U)} L(u) y_u$$
$$\overset{\text{Def. 8 point 2}}{\le} \sum_{u \in \mathcal{F} : d_G(u, N_G(U)) \le r} L(u) = \sum_{u \in \mathcal{F} : d_G(u,U) \le r+1} L(u)\,.$$

Together these equalities conclude the proof. ◀

We proceed with a pair of simple properties of transfers.

▶ **Fact 10.** *Let $G = (V,E)$ be a graph with $W \subseteq V$ and a capacity function $L : W \to \mathbb{Z}_{\ge 0}$, and let $y, y', y'' \in \mathbb{R}^W_{\ge 0}$. Assume y' is a distance-r transfer of (G, L, y) and y'' is a distance-r' transfer of (G, L, y'). Then y'' is a distance-$r + r'$ transfer of (G, L, y).*

▶ **Fact 11.** *Let $G = (V,E)$ and $G' = (V', E')$ be graphs with $W \subseteq V$ and $W \subseteq V'$ and a capacity function $L : W \to \mathbb{Z}_{\ge 0}$. Let $y, y' \in \mathbb{R}^W_{\ge 0}$ and let $f : \mathbb{Z}_{\ge 0} \to \mathbb{Z}_{\ge 0}$ be a monotonic function such that $d_G(u,v) \le f(d_{G'}(u,v))$ for any $u, v \in W$. Assume y' is a distance-r transfer of (G', L, y). Then y' is a distance-$f(r)$ transfer of (G, L, y).*

The following is the main technical contribution of [1].

▶ **Lemma 12** ([1])**.** *Let $T = (V,E)$ be a tree with a capacity function $L : V \to \mathbb{Z}_{\ge 0}$ and let $y \in [0,1]^V$ be a vector such that $y_v = 1$ for every non-leaf $v \in V$ and $\sum_{v \in V} y_v \in \mathbb{Z}_{\ge 0}$. Then one can find in polynomial time an integral distance-2 transfer of (T, L, y).*

6.2 Final rounding

▶ **Lemma 13.** *Let $G = (\mathcal{C}, \mathcal{F}, E)$ be a connected bipartite graph and let $S \subseteq \mathcal{F}$ such that $d(v,S) \le 5$ for every $v \in \mathcal{C} \cup \mathcal{F}$. There exists an auxiliary tree $T = (S, E_T)$ such that $d(u,u') \le 10$ for any $\{u,u'\} \in E_T$. Moreover, such a tree can be computed in polynomial time.*

Proof. We shall grow a tree adding a leaf in each step. At the beginning we select any $s \in S$ and initialize with a single-vertex tree. Assume we have already grown a tree with vertex-set $S' \subseteq S$. Choose a shortest path connecting S' to $S' \setminus S$. Such a path exists since G is connected. If its length is at most 10, we add the endpoint in $S \setminus S'$ to the tree,

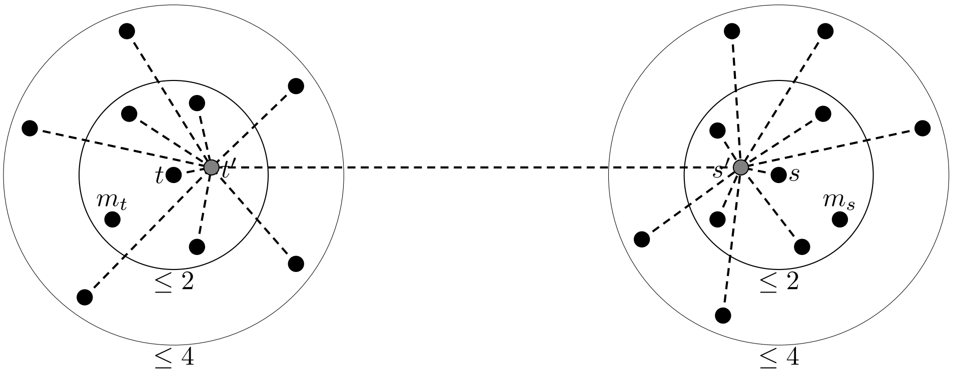

Figure 2 A fragment of the tree T' with $s, t \in S$. Nodes of $\mathcal{F}$ are marked in black, of S' in gray. Edges of T' are represented as dashed lines. Note that m_s and m_t are not vertices of T'.

joining it with the other endpoint. For a proof by contradiction assume that a shortest path has length greater than 10. Since G is bipartite, its length needs to be even, and thus at least 12. Choose the midpoint of such a path. Its distance both to S' and to $S' \setminus S$ is at least 6, otherwise the path could be shortened. This vertex contradicts the assumption that $d(v, S) \leq 5$ for every $v \in \mathcal{C} \cup \mathcal{F}$. ◀

We are ready to prove Lemma 7.

Proof of Lemma 7. Since G is connected and every vertex of G is within distance 5 from S, we can use Lemma 13 to construct a tree $T = (S, E_T)$. Let us add a duplicate s' of every $s \in S$ to create a bipartite graph $G' = (\mathcal{C}, \mathcal{F}', E')$, where $\mathcal{F}' = \mathcal{F} \cup S'$ and $S' = \{s' : s \in S\}$. For each $s \in S$ choose $m_s = \operatorname{argmax}\{L(u) : u \in N^2[s] \cap \mathcal{F}\}$ and set $L(s') = L(m_s)$. Let us create a tree T' with $V(T') = \mathcal{F}' \setminus \{m_s : s \in S\}$. We build it in two steps, see also Fig. 2:

1. create a tree with vertex set S' so that $\{u', v'\}$ is an edge iff $\{u, v\} \in E(T)$,
2. connect each vertex in $\mathcal{F} \setminus \{m_s : s \in S\}$ to the closest vertex in S'.

Observe that endpoints of the edges created in the first step are at most at distance 10 in G', while endpoints of the edges created in the second step, at most at distance 4. Consequently, $d_{G'}(u, v) \leq 10 d_{T'}(u, v)$ for any $u, v \in V(T')$. Moreover, note that all non-leaves of T' belong to S'.

Let (x, y) be a feasible solution of $LP_{k,p}(G, L, S)$. Note that y can be interpreted as a vector in $\mathbb{R}^{\mathcal{F}'}_{\geq 0}$ extending with zeroes at S'. We shall give an integral distance-24 transfer F of (G', L, y). Despite it being formally a transfer in G', F will be a subset of $\mathcal{F}$, i.e. a transfer of (G, L, y) as well.

Recall that by (2), the sets $N^2[s]$ are pairwise disjoint and in particular m_s are pairwise different. This lets us use (6) to gather in s' one unit from $N^2[s]$ for every $s \in S$ so that the whole value in m_s is transferred to s'. Note that $L(s') \geq L(u)$ for each $u \in N^2[s]$, so this way we obtain a distance-2 transfer y' of (G', L, y). Additionally, we have made sure that $y'_{m_s} = 0$, so y' can be interpreted as a vector in $\mathbb{R}^{V(T')}_{\geq 0}$, and that $y'_{s'} = 1$, so y' is 1 for all non-leaves of T'. This lets us use Lemma 12 to obtain an integral distance-2 transfer $F' \subseteq V(T')$ of (T', L, y'). According to Fact 11 it can be interpreted as a distance-20 transfer of (G', L, y'). Finally we move the value from s' to m_s for each $s \in S$. Note that these vertices have equal capacities, so this step can be interpreted as an integral distance-2 transfer.

The final transfer is therefore a composition of a distance-2 transfer, a distance-20 transfer and a distance-2 transfer. Thus, by Fact 10 it is a distance-24 transfer.[4] By Lemma 9 having an integral distance-24 transfer is enough to construct a distance-25 solution ϕ in polynomial time, which concludes the proof of Lemma 7. ◀

7 Wrap-up

With the results of previous section, we are ready to the prove the main theorem.

▶ **Theorem 14.** *The* Capacitated k-supplier with Outliers *problem admits a 25-approximation algorithm.*

Proof. Section 3 with Algorithm 1 provides (a Turing-like) reduction to graphic instances. Algorithm 2 of Section 4 given such an instance outputs several sets. Provided that a distance-1 solution exists, one of them is guaranteed to be a skeleton. Each of these sets is then processed separately. As described in Section 5, some redundant vertices are removed and the graph is partitioned into connected components. Dynamic programming (Lemma 6) is then used to find a compatible partition of k and p, so that each linear program $LP_{k_i,p_i}(G_i, L, S_i)$ admits a feasible solution. While this procedure might fail in general, it is guaranteed to succeed for a skeleton, hence at least once if a distance-1 solution exists.

Note that if such a partition is found, then for each of the instances (G_i, L, k_i, p_i) together with sets S_i, we can use Lemma 7 as all the conditions $(i) - (iv)$ are satisfied. A sum of solutions for these ℓ instances is finally returned as a distance-25 solution for the original graphic instance. ◀

Acknowledgements. We would like to thank Samir Khuller for suggesting the study of this variant of the k-center problem and helpful discussions.

References

1 Hyung-Chan An, Aditya Bhaskara, and Ola Svensson. Centrality of trees for capacitated k-center. *CoRR*, abs/1304.2983, 2013.

2 Vijay Arya, Naveen Garg, Rohit Khandekar, Adam Meyerson, Kamesh Munagala, and Vinayaka Pandit. Local search heuristic for k-median and facility location problems. In Jeffrey Scott Vitter, Paul G. Spirakis, and Mihalis Yannakakis, editors, *STOC*, pages 21–29. ACM, 2001.

3 Manisha Bansal, Naveen Garg, and Neelima Gupta. A 5-approximation for capacitated facility location. In Leah Epstein and Paolo Ferragina, editors, *ESA*, volume 7501 of *Lecture Notes in Computer Science*, pages 133–144. Springer, 2012.

4 Judit Bar-Ilan, Guy Kortsarz, and David Peleg. How to allocate network centers. *Journal of Algorithms*, 15(3):385–415, 1993.

5 Yair Bartal, Moses Charikar, and Danny Raz. Approximating min-sum k-clustering in metric spaces. In Jeffrey Scott Vitter, Paul G. Spirakis, and Mihalis Yannakakis, editors, *STOC*, pages 11–20. ACM, 2001.

[4] A simpler construction gives a distance-30 transfer, without introducing additional vertices S'. It is enough first to gather one unit from $N^2[s]$ in m_s and build a tree on vertices m_s, where adjacent vertices of the tree are at distance at most 14 in G. By using Lemma 12 one obtains a distance-28 transfer, which together with the initial distance-2 transfer gives an integral distance-30 transfer.

6 Jaroslaw Byrka and Karen Aardal. An optimal bifactor approximation algorithm for the metric uncapacitated facility location problem. *SIAM J. Comput.*, 39(6):2212–2231, 2010.

7 Moses Charikar, Sudipto Guha, Éva Tardos, and David B. Shmoys. A constant-factor approximation algorithm for the k-median problem. *Journal of Computer and System Sciences*, 65(1):129–149, 2002.

8 Moses Charikar, Samir Khuller, David M. Mount, and Giri Narasimhan. Algorithms for facility location problems with outliers. In S. Rao Kosaraju, editor, *SODA*, pages 642–651. ACM/SIAM, 2001.

9 Julia Chuzhoy, Sudipto Guha, Eran Halperin, Sanjeev Khanna, Guy Kortsarz, Robert Krauthgamer, and Joseph Naor. Asymmetric k-center is $\log^* n$-hard to approximate. *Journal of the ACM*, 52(4):538–551, 2005.

10 Julia Chuzhoy and Yuval Rabani. Approximating k-median with non-uniform capacities. In *SODA*, pages 952–958. SIAM, 2005.

11 Marek Cygan, MohammadTaghi Hajiaghayi, and Samir Khuller. LP rounding for k-centers with non-uniform hard capacities. In *FOCS*, pages 273–282. IEEE Computer Society, 2012.

12 M. R. Garey and David S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness.* W. H. Freeman, 1979.

13 Teofilo F. Gonzalez. Clustering to minimize the maximum intercluster distance. *Theoretical Computer Science*, 38:293–306, 1985.

14 Sudipto Guha and Samir Khuller. Greedy strikes back: Improved facility location algorithms. *Journal of Algorithms*, 31(1):228–248, 1999.

15 Dorit S. Hochbaum and David B. Shmoys. A best possible heuristic for the k-center problem. *Mathematics of Operations Research*, 10:180–184, 1985.

16 Dorit S. Hochbaum and David B. Shmoys. A unified approach to approximation algorithms for bottleneck problems. *Journal of the ACM*, 33(3):533–550, 1986.

17 Wen-Lian Hsu and George L. Nemhauser. Easy and hard bottleneck location problems. *Discrete Applied Mathematics*, 1:209–216, 1979.

18 Kamal Jain and Vijay V. Vazirani. Approximation algorithms for metric facility location and k-median problems using the primal-dual schema and lagrangian relaxation. *Journal of the ACM*, 48(2):274–296, 2001.

19 Samir Khuller and Yoram J. Sussmann. The capacitated k-center problem. *SIAM Journal on Discrete Mathematics*, 13(3):403–418, 2000.

20 Retsef Levi, David B. Shmoys, and Chaitanya Swamy. LP-based approximation algorithms for capacitated facility location. *Mathematical Programming*, 131(1-2):365–379, 2012.

21 Shi Li. A 1.488 approximation algorithm for the uncapacitated facility location problem. *Information and Computation*, 222:45–58, 2013.

22 Shi Li and Ola Svensson. Approximating k-median via pseudo-approximation. In Dan Boneh, Tim Roughgarden, and Joan Feigenbaum, editors, *STOC*, pages 901–910. ACM, 2013.

23 David B. Shmoys, Éva Tardos, and Karen Aardal. Approximation algorithms for facility location problems. In Frank Thomson Leighton and Peter W. Shor, editors, *STOC*, pages 265–274. ACM, 1997.

24 David P. Williamson and David B. Shmoys. *The Design of Approximation Algorithms.* Cambridge University Press, 2011.