Capacitated Center Problems with Two-Sided Bounds and Outliers

Hu Ding[1], Lunjia Hu[2], Lingxiao Huang[2] and Jian Li[2]

[1] Computer Science and Engineering, Michigan State University, East Lansing, MI, USA *

[2] Institute for Interdisciplinary Information Sciences, Tsinghua University, Beijing, China**

Abstract. In recent years, the capacitated center problems have attracted a lot of research interest. Given a set of vertices V, we want to find a subset of vertices S, called centers, such that the maximum cluster radius is minimized. Moreover, each center in S should satisfy some capacity constraint, which could be an upper or lower bound on the number of vertices it can serve. Capacitated k-center problems with one-sided bounds (upper or lower) have been well studied in previous work, and a constant factor approximation was obtained.

We are the first to study the capacitated center problem with both capacity lower and upper bounds (with or without outliers). We assume each vertex has a uniform lower bound and a non-uniform upper bound. For the case of opening exactly k centers, we note that a generalization of a recent LP approach can achieve constant factor approximation algorithms for our problems. Our main contribution is a simple combinatorial algorithm for the case where there is no cardinality constraint on the number of open centers. Our combinatorial algorithm is simpler and achieves better constant approximation factor compared to the LP approach.

1 Introduction

The k-center clustering is a fundamental problem in theoretical computer science and has numerous applications in a variety of fields. Roughly speaking, given a metric space containing a set of vertices, the k-center problem asks for a subset of k vertices, called centers, such that the maximum radius of the induced k clusters is minimized. Actually k-center clustering falls in the umbrella of the general *facility location* problems which have been extensively studied in the past decades. Many operation and management problems can be modeled as facility location problems, and usually the input vertices and selected centers are also called “clients” and “facilities” respectively. In this paper, we consider a

* HD is supported by the start-up fund from Michigan State University.

** LJH, LXH and JL are supported in part by the National Basic Research Program of China grants 2015CB358700, 2011CBA00300, 2011CBA00301, and the National NSFC grants 61033001, 61632016, 61361136003.

© Springer International Publishing AG 2017

F. Ellen et al. (Eds.): WADS 2017, LNCS 10389, pp. 325–336, 2017.
DOI: 10.1007/978-3-319-62127-2_28

significant generalization of the k-center problem, where each vertex is associated with a capacity interval; that is, the cardinality of the resulting cluster centered at the vertex should satisfy the given lower and upper capacity bounds (the formal definition is shown in Section 1.2). In addition, we also consider the case where a given number of vertices may be excluded as outliers.

Besides being a natural combinatorial problem on its own, the k-center problem with both capacity upper and lower bounds is also strongly motivated by several realistic issues raised in a variety of application contexts.

1. In the context of facility location, each open facility may be constrained by the maximum number of clients it can serve. The capacity lower bounds also come naturally, since an open facility needs to serve at least a certain number of clients in order to generate profit.
2. Several variants of the k-center clustering have been used in the context of preserving privacy in publication of sensitive data (see e.g., [1, 23, 26]). In such applications, it is important to have an appropriate lower bound for the cluster sizes, in order to protect the privacy to certain extent (roughly speaking, it would be relatively easier for an adversary to identify the clients inside a too small cluster).
3. Consider the scenario where the data is distributed over the nodes in a large network. We would like to choose k nodes as central servers, and aggregate the information of the entire network. We need to minimize the delay (i.e., minimize the cluster radius), and at the same time consider the balancedness, for the obvious reason that the machines receiving too much data could be the bottleneck of the system and the ones receiving too little data is not sufficiently energy-efficient [11].

Our problem generalizes the classic k-center problem as well as many important variants studied by previous authors. The optimal approximation results for the classic k-center problem appeared in the 80's: Gonzalez [15] and Hochbaum and Shmoys [17] provided a 2-approximation in a metric graph; moreover, they proved that any approximation ratio $c < 2$ would imply $P = NP$. The first study on capacitated (with only upper bounds) k-center clustering is due to Bar-Ilan et al. [5] who provided a 10-approximation algorithm for uniform capacities (i.e., all the upper bounds are identical). Further, Khuller and Sussmann [20] improved the approximation ratio to be 6 and 5 for hard and soft uniform capacities, respectively. [3] The recent breakthrough for non-uniform (upper) capacities is due to Cygan et al. [10]. They developed the first constant approximation algorithm based on LP rounding, though their approximation ratio is about hundreds. Following this work, An et al. [3] provided an approximation algorithm with the much lower approximation ratio 9. On the imapproximability side, it is impossible to achieve an approximation ratio lower than 3 for non-uniform capacities unless $P = NP$ [10].

[3] We can open more than one copies of a facility in the same node in the soft capacity version. But in the hard capacity version, we can only open at most one copy.

For the ordinary k-center with outliers, a 3-approximation algorithm was obtained by Charikar et al. [8]. Kociumaka and Cygan [21] studied k-center with non-uniform upper capacities and outliers, and provided a 25-approximation algorithm.

k-center clustering with lower bounds on cluster sizes was first studied in the context of privacy-preserving data management [26]. Aggarwal et al. [1] provided a 2-approximation and a 4-approximation for the cases without and with outliers, respectively. Further, Ene et al. [13] presented a near linear time $(4+\varepsilon)$-approximation algorithm in constant dimensional Euclidean space. Note that both [1, 13] are only for uniform lower bounds. Recently, Ahmadian and Swamy [2] provided a 3-approximation and a 5-approximation for the non-uniform lower bound case without and with outliers.

Our main results. To the best of our knowledge, we are the first to study the capacitated center with both capacity lower and upper bounds (with or without outliers). Recently, Ding [12] also studies k-center clustering with two-sided bounds in high dimension or any metric space when k is a constant, and provides a nearly linear time 4-approximation. Given a set V of n vertices, we focus on the case where the capacity of each vertex $u \in V$ has a uniform lower bound $L_u = L$ and a non-uniform upper bound U_u. Sometimes, we consider a generalized supplier version where we are only allowed to open centers among a facility set $\mathcal{F}$, see Definition 1 for details. We mainly provide first constant factor approximation algorithms for the following variants, see Table 1 for other results. Due to the lack of space, we defer many details and proofs to a full version.

1. (L,U,soft-$\emptyset$,p)-Center (Section 2.2): In this problem, both the lower bounds and the upper bounds are uniform, i.e., $L_u = L, U_u = U$ for all $u \in V$. The number of open centers can be arbitrary, i.e., there is no requirement to choose exactly k open centers. Moreover, we allow multiple open centers at a single vertex $u \in V$ (i.e., soft capacity). We may exclude $n-p$ outliers. We provide the first polynomial time combinatorial algorithm which can achieve an approximate factor of 5.
2. (L,$\{U_u\}$,$\emptyset$,p)-Center(Section 2.3): In this problem, the lower bounds are uniform, i.e., $L_u = L$ for all $u \in V$, but the upper bound can be nonuniform. The number of open centers can be arbitrary. We may exclude $n-p$ outliers. We provide the first polynomial time combinatorial 11-approximation for this problem.
3. (L,$\{U_u\}$,k)-Center (Section 3.3): In this problem, we would like to open exactly k centers, such that the maximum cluster radius is minimized. All vertices have the same capacity lower bounds, i.e., $L_u = L$ for all $u \in V$. But the capacity upper bounds may be nonuniform, i.e., each vertex u has an individual capacity upper bound U_u. Moreover, we do not exclude any outlier. We provide the first polynomial time 9-approximation algorithm for this problem, based on LP rounding.
4. (L,$\{U_u\}$,k,p)-Center (Section 3.3): This problem is the outlier version of the (L,$\{U_u\}$,k)-Center problem. The problem setting is exactly the same

except that we can exclude $n-p$ vertices as outliers. We provide a polynomial time 25-approximation algorithm for this problem.

Problem Setting		Approximation Ratio	
		Center Version	Supplier Version
Without k Constraint	$(L,U,\text{soft-}\emptyset,p)$	5	5
	$(L,U,\emptyset,p)$	10	23
	$(L,\{U_u\},\text{soft-}\emptyset,p)$	11	11
	$(L,\{U_u\},\emptyset,p)$	11	25
With k Constraint	(L,U,k)	6	9
	$(L,\{U_u\},k)$	9	13
	$(L,U,\text{soft-}k,p)$	13	13
	(L,U,k,p)	23	23
	$(L,\{U_u\},\text{soft-}k,p)$	25	25
	$(L,\{U_u\},k,p)$	25	25

Table 1. A summarization table for our results in this paper.

Our main techniques. In Section 2, we consider the first two variants which allow to open arbitrarily many centers. We design simple and faster combinatorial algorithms which can achieve better constant approximation ratios compared to the LP approach. For the simpler case $(L,U,\text{soft-}\emptyset,p)$-Center, we construct a data structure for all possible open centers. We call it a *core-center tree* (CCT). Our greedy algorithm mainly contains two procedures. The first procedure *pass-up* greedily assigns vertices to open centers from the leaves of CCT to the root. After this procedure, there may exist some unassigned vertices around the root. We then introduce the second procedure called *pass-down*, which assigns these vertices in order by finding an *exchange route* each time. For the more general case $(L,\{U_u\},\emptyset,p)$-Center, our greedy algorithm is similar but somewhat more subtle. We still construct a CCT and run the pass-up procedure. Then we obtain an open center set F, which may contain redundant centers. However, since we deal with hard capacities and outliers, we need to find a non-redundant open center set which is not 'too far' from F (see Section 2.3 for details) and have enough total capacities. Then by a pass-down procedure, we can assign enough vertices to their nearby open centers.

In Section 3 and 3.3, we consider the last two variants which require to open exactly k centers. We generalized the LP approach developed for k-center with only capacity upper bounds [3, 21] and obtain constant approximation schemes for two-sided capacitated bounds. The omitted proofs can be found in the full version in this paper.

1.1 Other Related Work

The classic k-center problem is quite fundamental and has been generalized in many ways, to incorporate various constraints motivated by different application scenarios. Recently, Fernandes et al. [14] also provided constant approximations for the fault-tolerant capacitated k-center clustering. Chen et al. [9] studied the matroid center problem where the selected centers must form an independent set of a given matroid, and provided constant factor approximation algorithms (with or without outliers).

There is a large body of work on approximation algorithms for the facility location and k-median problems (see e.g., [4, 6, 7, 16, 18, 19, 22, 24, 25]). Moreover, Dick et al. [11] studied multiple balanced clustering problems with uniform capacity intervals, that is, all the lower (upper) bounds are identical; they also consider the problems under the stability assumption.

1.2 Preliminaries

In this paper, we usually work with the following more general problem, called the capacitated k-supplier problem. It is easy to see it generalizes the capacitated k-center problem since we can not open centers at any vertex. The formal definition is as follows.

Definition 1. *(Capacitated k-supplier with two-sided bounds and outliers) Suppose that we have*

1. *Two integers $k, p \in \mathbb{Z}_{\geq 0}$;*
2. *A finite set $\mathcal{C}$ of clients, and a finite set $\mathcal{F}$ of facilities;*
3. *A symmetric distance function $\mathrm{d} : (\mathcal{C} \cup \mathcal{F}) \times (\mathcal{C} \cup \mathcal{F}) \to \mathbb{R}_{\geq 0}$ satisfying the triangle inequality;*
4. *A capacity interval $[L_u, U_u]$ for each facility $u \in \mathcal{F}$, where $L_u, U_u \in \mathbb{Z}_{\geq 0}$ and $L_u \leq U_u$.*

Our goal is to find a client set $C \subseteq \mathcal{C}$ of size at least p, an open facility set $F \subseteq \mathcal{F}$ of size exactly k, and a function $\phi : C \to F$ satisfying that $L_u \leq |\phi^{-1}(u)| \leq U_u$ for each $u \in F$, which minimize the maximum cluster radius $\max_{v \in C} \mathrm{d}(v, \phi(v))$. If the maximum cluster radius is at most r, we call the tuple (C, F, ϕ) a distance-r solution.

By the similar approach of Cygan et al. [21], we can reduce the $(\{L_u\},\{U_u\},k,p)$-Supplier problem to a simpler case. We first introduce some definitions.

Definition 2. *(Induced distance function) We say the distance function $\mathrm{d}_G : (\mathcal{C} \cup \mathcal{F}) \times (\mathcal{C} \cup \mathcal{F}) \to \mathbb{R}_{\geq 0}$ is induced by an undirected unweighted connected graph $G = (\mathcal{C} \cup \mathcal{F}, E)$ if*

1. *$\forall (u, v) \in E$, we have $u \in \mathcal{F}$ and $v \in \mathcal{C}$.*
2. *$\forall a_1, a_2 \in \mathcal{C} \cup \mathcal{F}$, the distance $\mathrm{d}_G(a_1, a_2)$ between a_1 and a_2 equals to the length of the shortest path from a_1 to a_2.*

Definition 3. *(Induced $(\{L_u\},\{U_u\},k,p)$-*Supplier *instance) An $(\{L_u\},\{U_u\},k,p)$-*Supplier *instance is called an induced $(\{L_u\},\{U_u\},k,p)$-*Supplier *instance if the following properties are satisfied:*

1. *The distance function d_G is induced by an undirected connected graph $G = (\mathcal{C} \cup \mathcal{F}, E)$.*
2. *The optimal capacitated k-supplier value is at most 1.*

Moreover, we say this instance is induced by G.

When the graph of interest G is clear from the context, we will use d instead of d_G for convenience. We then show a reduction from solving the generalized $(\{L_u\},\{U_u\},k,p)$-Supplier problem to solving induced $(\{L_u\},\{U_u\},k,p)$-Supplier instances by Lemma 1.

Lemma 1. *Suppose we have a polynomial time algorithm A that takes as input any induced $(\{L_u\},\{U_u\},k,p)$-Supplier instance, and outputs a distance-ρ solution. Then, there exists a ρ-approximation algorithm for the $(\{L_u\},\{U_u\},k,p)$-Supplier problem with polynomial running time.*

2 Capacitated Center with Two-Sided Bounds and Outliers

In this section, we consider the version that the number of open centers can be arbitrary. By the LP approach in Section 3.3 and enumerating the number of open centers, we can achieve approximation algorithms for different variants in this case. However, the approximation factor is not small enough. In this section, we introduce a new greedy approach in order to achieve better approximation factors. Since our algorithm is combinatorial, it is easier to be implemented and saves the running time compared to the LP approach.

2.1 Core-center tree (CCT)

Consider the $(L,\{U_u\},\emptyset,p)$-Supplier problem. By Lemma 1, we only need to consider induced $(L,\{U_u\},\emptyset,p)$-Supplier instances induced by an undirected unweighted connected graph $G = (\mathcal{C} \cup \mathcal{F}, E)$. We first propose a new data structure called *core-center tree (*CCT*)* as follows.

Definition 4. *(Core-center tree (*CCT*)) Given an induced $(L,\{U_u\},\emptyset,p)$-*Supplier *instance induced by an undirected unweighted connected graph $G = (\mathcal{C} \cup \mathcal{F}, E)$, we call a tree $T = (\mathcal{F}, E_T)$ a core-center tree(*CCT*) if the following properties hold.*

1. *For each edge $(u, u') \in E_T$, we have $\mathrm{d}_G(u, u') \leq 2$;*
2. *Suppose the root of T is at layer 0. Denote I to be the set of vertices in the even layers of T. We call I the core-center set of T. For any two distinct vertices $u, u' \in I$, we have $\mathrm{d}_G(u, u') \geq 3$.*

Lemma 2. *Given an induced $(L,\{U_u\},\emptyset,p)$-*Supplier *instance induced by an undirected unweighted connected graph $G = (\mathcal{C} \cup \mathcal{F}, E)$, we can construct a* CCT *in polynomial time.*

For any $u \in \mathcal{F}$, denote $N_G[u] = \{v \in \mathcal{C} : (u, v) \in E\}$ to be the collection of all neighbors of $u \in \mathcal{F}$. [4] W.l.o.g., we assume that $U_u \leq |N_G(u)|$ for every facility $u \in \mathcal{F}$ in this section. In fact, we can directly delete all $u \in \mathcal{F}$ satisfying that $|N_G[u]| < L$ from the facility set $\mathcal{F}$, since u can not be open in any optimal feasible solution. [5] Otherwise if $L \leq |N_G[u]| < U_u$, we set

[4] If $u \in \mathcal{C}$ is also a client, then $u \in N_G[u]$.

[5] If this deletion causes the induced graph unconnected, similar to Lemma 6 in [21], we divide the graph into different connected components, and consider each smaller induced instance based on different connected components.

$U_u \leftarrow \min\{U_u, |N_G[u]|\}$, which has no influence on any optimal feasible solution of the induced $(L,\{U_u\},\emptyset,p)$-SUPPLIER instance. The following lemma gives a useful property of CCT.

Lemma 3. *Given an induced $(L,\{U_u\},\emptyset,p)$-*SUPPLIER *instance induced by an undirected unweighted connected graph $G = (\mathcal{C} \cup \mathcal{F}, E)$, and a core-center tree $T = (\mathcal{F}, E_T)$, suppose I is the core-center set of T. Then, we can construct a function $\xi : \mathcal{C} \to \mathcal{F}$ satisfying the following properties in polynomial time.*

1. *For all $v \in \mathcal{C}$, we have $(\xi(v), v) \in E$;*
2. *For all $u \in I$, we have $|\xi^{-1}(u)| \geq L$.*

2.2 A Simple Case: $(L,U,$soft-$\emptyset,p)$-Supplier

We first consider a simple case where the capacity bounds (upper and lower) are uniform and soft. In this setting, we want to find an open facility set $F = \{u_i \mid u_i \in \mathcal{F}\}_i$. Note that we allow multiple open centers in F. We also need to find an assignment function $\phi : \mathcal{C} \to F$, representing that we assign every client $v \in \mathcal{C}$ to facility $\phi(v)$. The main theorem is as follows.

Theorem 1. *(main theorem) There exists a 5-approximation polynomial time algorithm for the $(L,U,$soft-$\emptyset,p)$-*SUPPLIER *problem.*

By Lemma 1, we only consider induced $(L,U,$soft-$\emptyset,p)$-SUPPLIER instances. Given an induced $(L,U,$soft-$\emptyset,p)$-SUPPLIER instance induced by an undirected unweighted connected graph $G = (\mathcal{C} \cup \mathcal{F}, E)$, recall that we can assume $|N_G[u]| \geq U_u \geq L$ for each $u \in \mathcal{F}$. We first construct a CCT $T = (\mathcal{F}, E_T)$ rooted at node u^*, and a function $\xi : \mathcal{C} \to \mathcal{F}$ satisfying Lemma 3. For a facility set $P \subseteq \mathcal{F}$, we denote $\xi^{-1}(P) = \bigcup_{u \in P} \xi^{-1}(u)$ to be the collection of clients assigned to some facility in P by ξ.

Our algorithm mainly includes two procedures. The first procedure is called *pass-up*, which is a greedy algorithm to map clients to facilities from the leaves of T to the root. After the 'pass-up' procedure, we still leave some unassigned clients nearby the root. Then we use a procedure called *pass-down* to allocate those unassigned clients by iteratively finding an *exchange route*. In the following, we give the details of both procedures.

Procedure Pass-Up. Assume that $|\mathcal{C}| = aL + b$ for some $a \in \mathbb{N}$ and $0 \leq b \leq L - 1$. In this procedure, we will find an open facility set F of size a. We also find an assignment function ϕ which assigns aL clients to some nearby facility in F except a client set $S \subseteq \mathcal{C}$. Here, S is a collection of b clients in $\xi^{-1}(u^*)$ nearby the root u^*. Our main idea is to open facility centers from the leaves of CCT T to the root iteratively. During opening centers, we assign exactly L 'close' clients to each center. Thus, there are b unassigned clients after the whole procedure.

We then describe an iteration of pass-up. Assume that I is the core-center set of T. At the beginning, we find a non-leaf vertex $u \in I$ satisfying that all of its grandchildren (if exists) are leaves. [6] We denote $P \subseteq \mathcal{F}$ to be the collection of all

[6] If multiple non-leaf nodes satisfy this property, we choose an arbitrary one.

children and all grandchildren of u. In the next step, we consider all unscanned clients in $\xi^{-1}(P)$, [7] and assign them to the facility u by ϕ. Note that we may open multiple centers at u. We want that each center at u serves exactly L centers. However, there may exist one center at u serving less than L unscanned clients in $\xi^{-1}(P)$. We assign some clients in $\xi^{-1}(u)$ to this center such that it also serves exactly L clients. After this iteration, we delete the subtree rooted at u from T except u itself.

Finally, the root u^* will become the only remaining node in T. We open multiple centers at u^*, each serving exactly L clients in $\xi^{-1}(u^*)$, until there are less than L unassigned clients. We denote S to be the collection of those unassigned clients. At the end of pass-up, we output an open facility set F, an unassigned client set S and an assigned function $\phi : \mathcal{C} \setminus S \to F$. We have the following lemma by the algorithm.

Lemma 4. *Given an induced (L,U,soft-∅,p)-*SUPPLIER *instance induced by an undirected unweighted connected graph $G = (\mathcal{C} \cup \mathcal{F}, E)$, assume that $|\mathcal{C}| = aL + b$ for some $a \in \mathbb{N}$ and $0 \le b \le L - 1$. The output of pass-up satisfies the following properties:*

1. *Each open facility $u_j \in F$ satisfies that $u_j \in I$, and $|F| = a$;*
2. *The unassigned client set $S \subseteq \xi^{-1}(u^*)$, and $|S| = b$;*
3. *For each facility $u_i \in F$, we have $|\phi^{-1}(u_i)| = L$.*
4. *For each client $v \in \mathcal{C} \setminus S$, $\phi(v)$ is either $\xi(v)$, or the parent of $\xi(v)$ in T, or the grandparent of $\xi(v)$ in T. Moreover, we have $\mathrm{d}_G(v, \phi(v)) \le 5$.*

Procedure Pass-Down. After the procedure *pass-up*, we still leave an unassigned client set S of size b. However, our goal is to serve at least p clients. Therefore, we need to modify the assignment function ϕ and serve more clients.

The procedure pass-down handles the remaining b clients in S one by one. At the beginning of pass-down, we initialize an 'unscanned' client set $B \leftarrow \mathcal{C} \setminus S$, i.e., B is the collection of those clients allowing to be reassigned by pass-down. In each iteration, we arbitrarily pick a client $v \in S$ and assign it to the root node u^*. However, if each open facility at u^* has already served U_{u^*} clients by ϕ, assigning v to u^* will violate the capacity upper bound. In this case, we actually find an open center $u_j \in F$ such that $|\phi^{-1}(u_j)| < U_j$, i.e., there are less than U_j clients assigned to u_j by ϕ. We then construct an *exchange route* consisting of open facilities in F. We first find a sequence of nodes $w_0 = u^*, w_1, \cdots, w_m = u_j$ in T satisfying that w_i is the grandparent of w_{i+1} in the core-center tree T for all $0 \le i \le m - 1$. Then for each node w_i $(1 \le i \le m - 1)$, we pick a client $v_i \in \xi^{-1}(w_i)$ which has not been reassigned so far. We call such a sequence of clients $v, v_1, \ldots, v_{m-1}$ an exchange route. Our algorithm is as follows: 1) we assign v to $\phi(v_1)$; 2) we iteratively reassign v_i to $\phi(v_{i+1})$ in order $(1 \le i \le m-2)$; 3) finally we reassign v_{m-1} to u_j. We then mark all clients v_i $(1 \le i \le m - 1)$ in the exchange route by removing them from the 'unscanned' client set B, and

[7] Here, unscanned clients are those clients that have not been assigned by ϕ before this iteration.

remove the client $\{v\}$ from the unassigned client set S. Note that our exchange route only increases the number of clients assigned to u_j by one. In fact, such an exchange route always exists in each iteration. Thus in each iteration, the procedure pass-down assigns one more client $v \in S$ to some open facility in F. At the end of pass-down, we output a client set $C \leftarrow \mathcal{C} \setminus S$ of size at least p, an open facility set F and an assigned function $\phi : C \to F$.

Now we prove the following lemma. Note that Theorem 1 can be directly obtained by Lemma 1 and Lemma 5.

Lemma 5. *The procedure pass-down outputs a distance-5 solution (C, F, ϕ) of the given induced $(L,U,\text{soft-}\emptyset,p)$-*SUPPLIER *instance induced by $G = (\mathcal{C} \cup \mathcal{F}, E)$ in polynomial time.*

2.3 $(L,\{U_u\},\emptyset,p)$-Center

In this subsection, we consider a more complicated case where the capacity upper bounds are non-uniform, and each vertex has a hard capacity.

Theorem 2. *(main theorem) There exists an 11-approximation polynomial time algorithm for the $(L,\{U_u\},\emptyset,p)$-*CENTER *problem.*

By Lemma 1, we only need to consider induced $(L,\{U_u\},\emptyset,p)$-SUPPLIER instances. For an induced $(L,\{U_u\},\emptyset,p)$-SUPPLIER instance induced by an undirected unweighted connected graph $G = (V = \mathcal{C} \cup \mathcal{F}, E)$, recall that we can assume $U_u \le |N_G(u)|$ for every vertex $u \in \mathcal{F}$. [8] Since we consider the center version, every vertex $v \in \mathcal{C}$ has an individual capacity interval $[L, U_v]$ and can be opened as a center as well.

Similar to $(L,U,\text{soft-}\emptyset,p)$-CENTER, our algorithm first computes a core-center tree $T = (\mathcal{F}, E)$ rooted at u^*, a core-center set I and a function ξ described as in Lemma 3. Assume that $|\mathcal{C}| = aL + b$ for some $a \in \mathbb{N}$ and $0 \le b \le L - 1$. We still use the procedure *pass-up* to compute an open set $F = \{u_1, u_2, \cdots, u_a\}$, an unassigned set $S \subseteq \xi^{-1}(u^*)$ of size $b < L$, and a function $\phi : (\mathcal{C} \setminus S) \to F$.

However, we can not apply *pass-down* directly since we consider non-uniform hard capacity upper bounds. Thus, we need the following lemma to modify the open center set F. We prove this lemma by Hall's theorem in the full version.

Lemma 6. *Given an induced $(L,\{U_u\},\emptyset,p)$-*CENTER *instance induced by $G = (V = \mathcal{C} \cup \mathcal{F}, E)$ where $|N_G(u)| \ge U_u$ for each $u \in \mathcal{F}$ and an open set $F = \{u_1, u_2, \cdots, u_a\}$ computed by* pass-up*, there exists a polynomial time algorithm that finds another open set $F' = \{u'_1, u'_2, \ldots, u'_a\}$ such that:*

1. *F' is a single set.*
2. *For all $1 \le i \le a$, we have $\mathrm{d}_G(u_i, u'_i) \le 6$.*
3. *$\sum_{i=1}^{a} U_{u'_i} \ge p$.*

---

[8] Recall that we may remove some facilities from $\mathcal{F}$ such that this assumption is satisfied. Thus, the set $\mathcal{F}$ may be a subset of V.

Proof of Theorem 2. By Lemma 6, we obtain another open set $F' = \{u'_1, u'_2, \ldots, u'_a\}$. We first modify U_{u_i} to be $U_{u'_i}$ for all $1 \le i \le a$. Then we apply the procedure *pass-down* according to the modified capacities. By Lemma 5, we obtain a distance-5 solution (C, F, ϕ). Since $\sum_{i=1}^{a} U_{u'_i} \ge p$, at least p vertices are served by ϕ. Finally, for each vertex $v \in C$ and $u_i \in F$ such that $\phi(v) = u_i$, we reassign v to $u'_i \in F'$, i.e., let $\phi(v) = u'_i$. By Lemma 6, we obtain a feasible solution for the given induced $(L,\{U_u\},\emptyset,p)$-CENTER instance. Since $\mathrm{d}(u_i, u'_i) \le 6$ $(1 \le i \le a)$, the capacitated center value of our solution is at most $5 + 6 = 11$. Combining with Lemma 1, we finish the proof.

3 Capacitated *k*-Center with Two-Sided Bounds and Outliers

Now we study the capacitated k-center problems with two-sided bounds. We consider that all vertices have a uniform capacity lower bound $L_v = L$, while the capacity upper bounds can be either uniform or non-uniform. Similar to [3, 21], we use the LP relaxation and the rounding procedure distance-r transfer.

3.1 LP Formulation

We first give a natural LP relaxation for $(\{L_u\},\{U_u\},k,p)$-SUPPLIER.

Definition 5. *($\mathsf{LP}_r(G)$) Given an $(\{L_u\},\{U_u\},k,p)$-*SUPPLIER *instance, the following feasibility $\mathsf{LP}_r(G)$ that fractionally verifies whether there exists a solution that assigns at least p clients to an open center of distance at most r:*

$$\boxed{\begin{array}{ll} 0 \le x_{uv}, y_u \le 1, & \forall u \in \mathcal{F}, v \in \mathcal{C}; \\ x_{uv} = 0, & \text{if } \mathrm{d}(u,v) > r; \\ x_{uv} \le y_u, & \forall u \in \mathcal{F}, v \in \mathcal{C}; \\ \sum_{u \in \mathcal{F}} y_u = k; & \\ \sum_{u \in \mathcal{F}, v \in \mathcal{C}} x_{uv} \ge p; & \\ \sum_{u \in \mathcal{F}} x_{uv} \le 1, & \forall v \in \mathcal{C}; \\ L_u y_u \le \sum_{v \in \mathcal{C}} x_{uv} \le U_u y_u, & \forall u \in \mathcal{F}. \end{array}}$$

Here we call x_{uv} an assignment variable representing the fractional amount of assignment from client v to center u, and y_u the opening variable of $u \in \mathcal{F}$. For convenience, we use x, y to represent $\{x_{uv}\}_{u \in \mathcal{F}, v \in \mathcal{C}}$ and $\{y_u\}_{u \in \mathcal{F}}$, respectively.

By Definition 3, $\mathsf{LP}_1(G)$ must have a feasible solution for any induced $(\{L_u\},\{U_u\},k,p)$-SUPPLIER instance. We recall a rounding procedure called distance-r transfer.

3.2 Distance-r Transfer

We first extend the definition of distance-r transfer proposed in [3, 21] by adding the third condition. For a vertex $a \in \mathcal{C} \cup \mathcal{F}$ and a set $B \subseteq \mathcal{C} \cup \mathcal{F}$, we define $\mathrm{d}(a, B) = \min_{b \in B} \mathrm{d}(a, b)$.

Definition 6. *Given an $(\{L_u\},\{U_u\},k,p)$-*SUPPLIER *instance and $y \in \mathbb{R}^{\mathcal{F}}_{\ge 0}$, a vector $y' \in \mathbb{R}^{\mathcal{F}}_{\ge 0}$ is a distance-r transfer of y if*

1. $\sum_{u \in \mathcal{F}} y'_u = \sum_{u \in \mathcal{F}} y_u$;

2. $\sum_{w\in\mathcal{F}:\mathrm{d}(w,W)\leq r} U_w y'_w \geq \sum_{u\in W} U_u y_u$ *for all* $W \subseteq \mathcal{F}$*;*
3. $\sum_{w\in\mathcal{F}:\mathrm{d}(w,W)\leq r} L_w y_w \geq \sum_{u\in W} L_u y'_u$ *for all* $W \subseteq \mathcal{F}$*.*

If y' *is a characteristic vector of* $F \subseteq \mathcal{F}$*, we say that* F *is an integral distance-r transfer of* y*.*

In this paper, we add the third condition to satisfy the capacity lower bounds. Like in [3, 21], we still have the following lemma.

Lemma 7. *Given an (*$\{L_u\}$*,*$\{U_u\}$*,k,p)-*Supplier *problem, assume* (x, y) *is a feasible solution of* $\mathsf{LP}_1(G)$ *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.*

3.3 Capacitated *k*-Center with Two-Sided Bounds and Outliers

Now we are ready to solve the (L,$\{U_u\}$,k,p)-Supplier problem. By Lemma 7, we only need to find an integral distance-r transfer satisfying Definition 6 given a feasible fractional solution (x, y) of $\mathsf{LP}_1(G)$. Fortunately, the rounding schemes in [3, 21] have this property. Thus, we have the following theorem by [3].

Theorem 3. *There is a polynomial time 9-approximation algorithm for the (L,*$\{U_u\}$*,k)-*Center *problem. For the uniform capacity upper bound version, the (L,U,k)-*Center *problem admits a 6-approximation.*

Theorem 4. *There is a polynomial time 13-approximation algorithm for the (L,*$\{U_u\}$*,k)-*Supplier *problem. For the uniform capacity upper bound version, the (L,U,k)-*Supplier *problem admits a 9-approximation.*

By [21], we have the following theorem.

Theorem 5. *There is a polynomial time 25-approximation algorithm for the (L,*$\{U_u\}$*,k,p)-*Supplier *problem and the (L,*$\{U_u\}$*,soft-k,p)-*Supplier *problem. For the uniform capacity upper bound version, the (L,U,k,p)-*Supplier *problem admits a 23-approximation, and the (L,U,soft-k,p)-*Supplier *problem admits a 13-approximation.*

References

1. Aggarwal, G., Panigrahy, R., Feder, T., Thomas, D., Kenthapadi, K., Khuller, S., Zhu, A.: Achieving anonymity via clustering. ACM Trans. Algorithms 6(3), 49:1–49:19 (2010)
2. Ahmadian, S., Swamy, C.: Approximation algorithms for clustering problems with lower bounds and outliers. arXiv preprint arXiv:1608.01700 (2016)
3. An, H.C., Bhaskara, A., Chekuri, C., Gupta, S., Madan, V., Svensson, O.: Centrality of trees for capacitated k-center. Mathematical Programming 154(1-2), 29–53 (2015)
4. Arya, V., Garg, N., Khandekar, R., Meyerson, A., Munagala, K., Pandit, V.: Local search heuristics for k-median and facility location problems. SIAM Journal on computing 33(3), 544–562 (2004)

5. Barilan, J., Kortsarz, G., Peleg, D.: How to allocate network centers. Journal of Algorithms 15(3), 385–415 (1993)
6. Charikar, M., Guha, S.: Improved combinatorial algorithms for facility location problems. SIAM Journal on Computing 34(4), 803–824 (2005)
7. Charikar, M., Guha, S., Tardos, É., Shmoys, D.B.: A constant-factor approximation algorithm for the k-median problem. In: Proceedings of the thirty-first annual ACM symposium on Theory of computing. pp. 1–10. ACM (1999)
8. Charikar, M., Khuller, S., Mount, D.M., Narasimhan, G.: Algorithms for facility location problems with outliers. In: Proceedings of the twelfth annual ACM-SIAM symposium on Discrete algorithms. pp. 642–651. Society for Industrial and Applied Mathematics (2001)
9. Chen, D.Z., Li, J., Liang, H., Wang, H.: Matroid and knapsack center problems. Algorithmica 75(1), 27–52 (2016)
10. Cygan, M., Hajiaghayi, M., Khuller, S.: Lp rounding for k-centers with non-uniform hard capacities. In: Proceedings of the 2012 IEEE 53rd Annual Symposium on Foundations of Computer Science. pp. 273–282. IEEE Computer Society (2012)
11. Dick, T., Li, M., Pillutla, V.K., White, C., Balcan, M.F., Smola, A.: Data driven resource allocation for distributed learning. arXiv preprint arXiv:1512.04848 (2015)
12. Ding, H.: Balanced k-center clustering when k is a constant. arXiv preprint arXiv:1704.02515 (2017)
13. Ene, A., Har-Peled, S., Raichel, B.: Fast clustering with lower bounds: No customer too far, no shop too small. arXiv preprint arXiv:1304.7318 (2013)
14. Fernandes, C.G., de Paula, S.P., Pedrosa, L.L.: Improved approximation algorithms for capacitated fault-tolerant k-center. In: Latin American Symposium on Theoretical Informatics. pp. 441–453. Springer (2016)
15. Gonzalez, T.F.: Clustering to minimize the maximum intercluster distance. Theoretical Computer Science 38, 293–306 (1985)
16. Guha, S., Khuller, S.: Greedy strikes back: Improved facility location algorithms. Journal of algorithms 31(1), 228–248 (1999)
17. Hochbaum, D.S., Shmoys, D.B.: A best possible heuristic for the k-center problem. Mathematics of operations research 10(2), 180–184 (1985)
18. Jain, K., Mahdian, M., Saberi, A.: A new greedy approach for facility location problems. In: Proceedings of the thiry-fourth annual ACM symposium on Theory of computing. pp. 731–740. ACM (2002)
19. Jain, K., Vazirani, V.V.: Approximation algorithms for metric facility location and k-median problems using the primal-dual schema and lagrangian relaxation. Journal of the ACM (JACM) 48(2), 274–296 (2001)
20. Khuller, S., Sussmann, Y.J.: The capacitated k-center problem. SIAM Journal on Discrete Mathematics 13(3), 403–418 (2000)
21. Kociumaka, T., Cygan, M.: Constant factor approximation for capacitated k-center with outliers. arXiv preprint arXiv:1401.2874 (2014)
22. Korupolu, M.R., Plaxton, C.G., Rajaraman, R.: Analysis of a local search heuristic for facility location problems. Journal of algorithms 37(1), 146–188 (2000)
23. Li, J., Yi, K., Zhang, Q.: Clustering with diversity. In: International Colloquium on Automata, Languages, and Programming. pp. 188–200. Springer (2010)
24. Li, S.: A 1.488 approximation algorithm for the uncapacitated facility location problem. Information and Computation 222, 45–58 (2013)
25. Li, S., Svensson, O.: Approximating k-median via pseudo-approximation. SIAM Journal on Computing 45(2), 530–547 (2016)
26. Sweeney, L.: k-anonymity: A model for protecting privacy. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems 10(05), 557–570 (2002)