The Heterogeneous Capacitated k-Center Problem

Deeparnab Chakrabarty
Microsoft Research India
deeparnab@gmail.com

Ravishankar Krishnaswamy
Microsoft Research India
ravishankar.k@gmail.com

Amit Kumar
Comp. Sci. & Engg., IIT Delhi
amitk@cse.iitd.ac.in

Abstract

The abstract goes here.

1 Introduction

The capacitated k-center problem is a classic optimization problem where a finite metric space (X, d) needs to be partitioned into k clusters so that every cluster has cardinality at most come specified value C, and the objective is to minimize the maximum intra-cluster distance. This problem introduced by Bar-Ilan et al [**?**] has many applications in **blah blah blah**. One application is deciding placement of machine locations (centers of clusters) in a network scheduling [**?**] environment where jobs arise in a metric space and the objective function has a job-communication (intra-cluster distance) and machine-load (cardinality) component.

Add Venkat's References?

The above problem is *homogeneous* in the sizes of the clusters, that is, it has the same cardinality constraint L for each cluster. In many applications, one would ask for a *heterogeneous* version of the problem where we have a different cardinality constraint for the clusters. For instance in the network scheduling application above, suppose we had machines of differing speeds. We could possibly load higher-speed machines with more jobs than lower-speed ones. In this paper, we study this heterogenous version.

Definition 1. (The Heterogeneous Cap-k-Center Problem.) *The input to this problem is a metric space* $(X = F \cup C, d)$ *and a collection of* heterogeneous *capacities:* $(k_1, c_1), (k_2, c_2), \ldots, (k_P, c_P)$. *The output is to open facilities* $F_1 \cup \cdots \cup F_P$ *in* F *such that* $|F_p| \leq k_p$, *and find an assignment* $\phi : C \to F_1 \cup F_2 \cup \cdots \cup F_P$ *such that* $|\{j \in C : \phi(j) \in F_p\}| \leq c_p$ *for all* $1 \leq p \leq P$. *The objective is to minimize* $\max_{j \in C} d(j, \phi(j))$. *We use* OPT *to denote this latter distance of the optimum solution.*

The Heterogeneous Cap-k-Center problem is relevant in many applications where the resources available are heterogenous. The machine placement problem was one example which has applications in network scheduling [**?**] and distributed databases [**?**, **?**]. Another example is that of vehicle routing problems [**?**] with different sized fleets. A third relevant application may be clustering; often clusters of equal sizes are undesirable [**?**] and explicitly introducing heterogeneous constraints might lead to desirable clusters. In this paper, we investigate the worst-case complexity of the problem.

Bar-Ilan et al [**?**] gave a 10-approximation for the homogeneous capacitated k-center problem which was improved to a 6-factor approximation by Khuller and Sussmann [**?**]. One cannot get a better than 3-approximation even for the *uncapacitated* k-center problem [**?**]. More recently, the *non-uniform* capacitated k-center problem was considered [**?**, **?**] in the literature: in this problem every facility $v \in F$ has a pre-determined capacity c_v if opened (and 0 otherwise). We remark that the non-uniform version and our heterogeneous version seem unrelated in the sense that none is a special case of the other. Cygan et al [**?**] gave a $O(1)$-approximation for the problem which was improved to a 11-approximation by An et al [**?**][1].

Connection to 3-Partitioning. To underscore the difficulty and difference of Heterogeneous Cap-k-Center from the homogeneous capacitated k-center problems, we relate it to the classic 3-Partitioning problem [**?**]: in this problem we are given $3t$ non-negative numbers $\{a_1, \ldots, a_{3t}\}$ summing to Dt, and we have to decide if there is a partition into t-groups $S_1, \ldots, S_t$ such that $|S_i| = 3$ and $\sum_{j \in S_i} a_j = D$ for all i.

Given an instance of 3-Partition, consider an instance of Heterogeneous Cap-k-Center as follows. We let $n_i = 1$ and $c_i = a_i$ for $1 \leq i \leq 3t$. Consider a metric space where $X = F \cup C$ where X is partitioned into $X_1 = (F_1 \cup C_1), \ldots, X_t = (F_t \cup C_t)$ such that $|F_i| = 3$ and $|C_i| = D$ for all i. Furthermore, for any pair of points u, v in the same X_i their distance is 0 and otherwise ∞. Now observe that OPT for the Heterogeneous Cap-k-Center instance is *finite* if and only if the 3-Partitioning instance is a Yes-instance. In other words, unless $P = NP$, there can be no approximation algorithm for the problem unless we allow some capacity violation. This phenomenon, which is in contrast to the homogeneous version, motivates us to look at bicriteria algorithms.

Definition 2. *An* (a, b)*-bicriteria approximation algorithm for* Heterogeneous Cap-k-Center *assigns locations to* a *centers at most* $a \cdot$ OPT *away, and each center of type* i *serves at most* $b \cdot c_i$ *clients.*

[1] talk about distinction between center and supplier

1.1 Our Results

Our main result is a constant factor approximation algorithm which violates the capacities by at most a constant factor.

Theorem 1.1. *There is a polynomial time $(O(1), O(1))$-bicriteria approximation algorithm for the Heterogeneous Cap-k-Center problem.*

Although the violation in the capacities is large (although bounded by a constant), it already implies $O(1)$-approximations for the machine placement problem alluded to in the first paragraph. This maybe of independent interest.

Definition 3. (Machine Placement Problem.) *The input is a metric space $(X = F \cup C, d)$ with jobs with processing times p_j at locations C. We are also given P machines with speeds $s_1, s_2, \ldots, s_P$. The goal is to find a placement of these machines on F and schedule the jobs on these machines so as to minimize the makespan. The completion time of a job equals the time to reach the machine plus the processing time.*

Any (a, b)-bicriteria approximation algorithm for Heterogeneous Cap-k-Center implies a $O(a + b)$ approximation for the machine placement problem. Therefore, from Theorem 1.1, we get the following.

Theorem 1.2. *There is a polynomial time $O(1)$-approximate algorithm for the machine placement problem.*

The reduction from 3-Partitioning does not rule out non-trivial approximation algorithms which violate the capacity by $(1+\epsilon)$-factor for arbitrarily small $\epsilon > 0$. Indeed, for 3-Partitioning, there exist algorithms [**?**] which either return No, or find a partition with $\sum_{j \in S_i} a_j \geq D(1 - \varepsilon)$. However, as we discuss below, many *cardinality constrained scheduling problems* are also special cases of the Heterogeneous Cap-k-Center problem. There are currently no PTASes known for these problems and in a sense which we elaborate in Section **??** capture the difficulty in the problem. Nevertheless, we are able to obtain non-trivial algorithms with $(1 + \epsilon)$-capacity violation.

Theorem 1.3. *For any $\varepsilon > 0$, there exists an $(O(\log n/\varepsilon), (1 + \varepsilon))$-bicriteria approximation algorithm for the Heterogeneous Cap-k-Center problem running in time $C_\varepsilon^{\tilde{O}(\log^3 n)}$ for a constant C_ϵ depending only on ϵ.*

Connection to Cardinality Constrained Scheduling. In the classic scheduling problem of makespan minimization with identical machines ($P||C_{max}$) one is given m machines and n jobs with processing times $(p_1, \ldots, p_n)$ and the objective is to schedule them so as to minimize the maximum load on a machine. In the closely related 'max-min' version the objective is to maximize the minimum loaded machine. Abusing Graham's notation, let us denote this problem as $P||C_{min}$. Both these problems, and their uniform speed versions $Q||C_{max}$ and $Q||C_{min}$, admit PTASes [**?**]. In the uniform speed versions, each machine i has a speed s_i and the processing time of job j on machine i is p_j/s_i.

In the cardinality constrained version, the problem furthermore specifies a positive integer k_i for each machine indicating the maximum number of jobs that can be scheduled on it. The min-max or makespan minimization version, denoted as $P|k_i|C_{max}$ is called the k_i-partitioning problem [**?**]. This extra constraint makes the problem harder as existing ideas for PTASes do not seem to work; the best known approximation factor for $P|k_i|C_{max}$ is 1.5 due to Kellerer and Kotov [**?**]. The max-min problem has not been investigated much (but see Section 1.3). It is not too hard to modify the reduction from 3-Partition (see Section **??**) to show that the Heterogeneous Cap-k-Center problem is as hard as $Q|k_i|C_{min}$, that is the capacity constrained max-min problem on *uniform speed* machines.

Theorem 1.3 is obtained by proving a kind of converse. We show that any α-approximation for the $Q|k_i|C_{min}$ problem gives a $(O(\log n/\epsilon), \alpha(1 + \varepsilon)))$-approximation for the Heterogeneous Cap-k-Center problem. Theorem 1.3 then follows from the following theorem.

Theorem 1.4. *There is a QPTAS for the $Q|k_i|C_{min}$ problem.*

1.2 Technical Discussion

We give a brief discussion of how we prove our main theorems (Theorem 1.3 and 1.1). We start by writing the natural assignment LP relaxation for the problem. As is usual, we guess OPT and scale everything such that $\mathsf{OPT} = 1$. We have opening variables y_{ip} for every $i \in F, p \in [P]$ indicating whether we open a facility with capacity c_p at location i. We have connection variables x_{ijp} indicating the fraction to which client $j \in C$ connects to a facility at location i where a type p facility has been opened. We force $x_{ijp} = 0$ for all $d(i,j) > 1$. If $\mathsf{OPT} = 1$, there is a feasible (x, y) solution to the following system of inequalities.

$$\forall j \in C, \quad \sum_{i \in F} \sum_{p \in [P]} x_{ijp} \geq 1 \quad (1)$$

$$\forall i \in F, p \in [P], \quad \sum_{j \in C} x_{ijp} \leq c_p y_{ip} \quad (2)$$

$$\forall p \in [P], \quad \sum_{q \geq p} y_{iq} \leq \sum_{q \geq p} k_q \quad (3)$$

$$\forall i \in F, j \in C, p \in [P], \quad x_{ijp} \leq y_{ip} \quad (4)$$

$$\forall i \in F, \quad \sum_{p \in [P]} y_{ip} \leq 1 \quad (5)$$

$$\forall i \in F, j \in C, p \in [P], \quad x_{ijp}, y_{ip} \geq 0 \quad (6)$$

For technical reasons, we have written (3) as a constraint on the prefix-sums rather than individual capacities. However, a feasible integral solution satisfying (3) can easily be converted to a feasible solution satisfying individual capacities.

The above LP has bad integrality gap even when we allow arbitrary violation of capacities. Consider the following instance. The metric space X is partitioned into $(F_1 \cup C_1) \cup \cdots \cup (F_K \cup C_K)$, with $|F_k| = 2$ and $|C_k| = K$ for all $1 \leq k \leq K$. The distance between any two points in $F_i \cup C_i$ is 1 for all i, while all other distances are ∞. The capacities available are $k_1 = K$ facilities with capacity $c_1 = 1$ and $k_2 = K - 1$ facilities with capacity $c_2 = K$. It is easy to see that integrally any solution would violate capacities by a factor of $K/2$. On the other hand, there is a feasible solution for the above LP relaxation: for $F_k = \{a_k, b_k\}$, we set $y_{a_k 2} = 1 - 1/K$ and $y_{b_k 1} = 1$, and for all $j \in C_k$, we set $x_{a_k j 2} = 1 - 1/K$ and $x_{b_k j 1} = 1/K$.

Decomposition Theorem. The integrality gap suggests we need to strengthen our LP, and we will indeed do so. However, the above LP suffices to tease out the instance into parts that are indeed "well roundable" and problematic parts which look like the example above. To make this precise, let us introduce two definitions.

- A subset of facilities S is said to be (a, b)-*roundable wrt a feasible solution* (x, y) to (1)-(6) if its diameter is at most a and there is a rounding $Y_{ip} \in \{0, 1\}$ for all $i \in S, p \in [P]$ such that the rounded solution satisfies cardinality constraints, and has enough capacity to satisfy $1/b$th of the *fractional* demand incident on S. In other words, if we install $b \cdot c_p$ capacity at the location where $Y_{ip} = 1$, then it can satisfy the fractional demand incident on S.
- A subset S of facilities is called a *complete neighborhood* if there exists some clients $J \subseteq C$ such that S contains all the facilities in distance one to clients in J. That is, for all $j \in J$, the only facilities i with $x_{ijp} > 0$ must lie in S. Note that in the integrality gap example above, the F_k's are all complete neighborhoods due to the client set C_k's. Also note that these complete neighborhood sets are also encountered in the reductions from 3-Partition and the cardinality constrained scheduling problems.

Note that if our instance can be partitioned into roundable sets then we would be done. If our instance consists of only complete neighborhood sets, then we could hope to use techniques developed for scheduling algorithms. However, a priori, the instances are a mixture of both. Our main technical hammer is the decomposition theorem (Theorem **??**) which says that given feasible solution (x, y) to (1)-(6), in polynomial time we can decompose the instance $(F \cup C, d)$ into two collections $\mathcal{S}$ and $\mathcal{T}$ where every set S in $\mathcal{S}$ is a $(\tilde{O}(1/\varepsilon), (1+\varepsilon))$-roundable set, and every set $T \in \mathcal{T}$ is a complete neighborhood set of diameter ≤ 4. This tells us that the core difficulty that the above LP faces are indeed complete neighborhood sets of small diameter, and shows us how to strengthen our LP relaxation. **Maybe add a few lines on how we obtain – connections to region growing algorithms.**

Next we describe two ways to strengthen the LP. One leads to a polynomial time algorjthm but gives $O(1)$-factor violation of the capacities, the other leads to a quasipolynomial time $(1+\varepsilon)$-factor violation in the capacities. Both LPs, put extra constraints on the vector of y_{ip}'s. Interestingly, both LPs have exponentially number of variables and constraints we do not know if either LP is polynomial or quasipolynomial time

solvable. Rather, we use the "round-and-cut" strategy [**?**] where the constraints+auxiliary variables are added in phases *only if* a rounding algorithm fails. This technique has led to many new algorithms in the recent past, and ours adds to this canon of growing work.

LP strengthening. One useful idea that has helped for scheduling problems [**?**] is to look at *configuration LP*. For our problem, we add the following constraint. For a subset of clients $J \subseteq C$ let $\Gamma(J) \subseteq F$ be the facilities at distance 1. Since J can only be assigned to clients in $\Gamma(J)$, there must be enough capacity installed on $\Gamma(J)$ – in particular, the multiset/configuration S of capacities must be of cardinality $\leq |\Gamma(J)|$ and total capacity $\geq |J|$. We strengthen our LP (1)-(6) by adding that for every $J \subseteq C$, the y_{ip} vector must *dominate* a feasible configuration LP solution for the complete neighborhood $(\Gamma(J), J)$.

As mentioned above, this huge LP is perhaps not solvable in polynomial time. Instead we add these constraints iteratively. In every phase, given a feasible (x, y) to (1)-(6), we apply our decomposition theorem and obtain the complete neighborhood sets and add the strengthened constraints for these sets. The analysis of the ellipsoid algorithm implies in polynomially many phases we will reach a solution (x, y) where the y's satisfy the configuration LP constraints for all the complete neighborhood sets. Our final contribution is an $O(1)$-rounding of the configuration LP. **deepc: maybe expand on this as well.**

1.3 Related Work and Open Problems

Work on non-uniform capacities

Work of Shi Li

Constrained Scheduling

Heterogeneous problems

Open questions