An Improved Approximation Algorithm for Knapsack Median Using Sparsification

**Jarosław Byrka · Thomas Pensyl ·
Bartosz Rybicki · Joachim Spoerhase ·
Aravind Srinivasan · Khoa Trinh**

Abstract Knapsack median is a generalization of the classic k-median problem in which we replace the cardinality constraint with a knapsack constraint. It is currently known to be 32-approximable. We improve on the best known algorithms in several ways, including adding randomization and applying sparsification as a preprocessing step. The latter improvement produces the first LP for this problem with bounded integrality gap. The new algorithm obtains an approximation factor of 17.46. We also give a 3.05 approximation with small budget violation.

Jarosław Byrka, Bartosz Rybicki and Joachim Spoerhase
Institute of Computer Science
University of Wroclaw, Poland
E-mail: {jby,bry}@cs.uni.wroc.pl, joachim.spoerhase@uni-wuerzburg.de

Thomas Pensyl and Khoa Trinh
Department of Computer Science
University of Maryland, College Park, MD 20742
E-mail: {tpensyl,khoa}@cs.umd.edu

Aravind Srinivasan
Department of Computer Science and Instute for Advanced Computer Studies
University of Maryland, College Park, MD 20742
E-mail: srin@cs.umd.edu

1 Introduction

k-MEDIAN is a classic problem in combinatorial optimization. Herein, we are given a set of clients $\mathcal{C}$, facilities $\mathcal{F}$, and a symmetric distance metric c on $\mathcal{C} \cup \mathcal{F}$. The goal is to open k facilities such that we minimize the total connection cost (distance to nearest open facility) of all clients. A natural generalization of k-MEDIAN is KNAPSACK MEDIAN (KM), in which we assign nonnegative weight w_i to each facility $i \in \mathcal{F}$, and instead of opening k facilities, we require that the sum of the open facility weights be within some budget B.

While KM is not known to be harder than k-MEDIAN, it has thus far proved more difficult to approximate. k-MEDIAN was first approximated within constant factor $6\frac{2}{3}$ in 1999[2], with a series of improvements leading to the current best-known factor of $2.674[1]^1$. KM was first studied in 2011 by Krishnaswamy et. al. [6], who gave a bicriteria $16 + \epsilon$ approximation which slightly violated the budget. Then Kumar gave the first true constant factor approximation for KM with factor 2700 [7], subsequently reduced to 34 by Charikar&Li[3] and then to 32 by Swamy[10].

This paper's algorithm has a flow similar to Swamy's: we first get a half-integral solution (except for a few 'bad' facilities), and then create pairs of half-facilities, opening one facility in each pair. By making several improvements, we reduce the approximation ratio to 17.46. The first improvement is a simple modification to the pairing process so that every half-facility is guaranteed either itself or its closest neighbor to be open (versus having to go through two 'jumps' to get to an open facility). The second improvement is to randomly sample the half-integral solution, and condition on the probability that any given facility is 'bad'. The algorithm can be derandomized with linear loss in the runtime.

The third improvement deals with the bad facilities which inevitabley arise due to the knapsack constraint. All previous algorithms used Kumar's bound from [7] to bound the cost of nearby clients when bad facilities must be closed. However, we show that by using a sparsification technique similar in spirit to - but distinct from - that used in [8], we can focus on a subinstance in which the connection costs of clients are guaranteed to be evenly distributed throughout the instance. This allows for a much stronger bound than Kumar's, and also results in an LP with bounded integrality gap, unlike previous algorithms.

Another alternative is to just open the few bad facilities and violate the budget by some small amount, as Krishnaswamy et. al. did when first introducing KM. By preprocessing, we can ensure this violates the budget by at most ϵB. We show that the bi-point solution based method from [8] can be adapted for KM using this budget-violating technique to get a 3.05 approximation.

---

[1] the paper claims 2.611, but a very recent correction changes this to 2.674.

1.1 Preliminaries

Let $n = |\mathcal{F}| + |\mathcal{C}|$ be the size of the instance. For the ease of analysis, we assume that each client has unit demand. (Indeed, our algorithm easily extends to the general case.) For a client j, the connection cost of j, denoted as $\text{cost}(j)$, is the distance from j to the nearest <u>open</u> facility in our solution. The goal is to <u>open</u> a subset $\mathcal{S} \subseteq \mathcal{F}$ of facilities such that the total connection cost is minimized, subject to the <u>knapsack constraint</u> $\sum_{i \in \mathcal{S}} w_i \leq B$.

The natural LP relaxation of this problem is as follows.

$$
\begin{aligned}
\text{minimize} \quad & \sum_{i \in \mathcal{F}, j \in \mathcal{C}} c_{ij} x_{ij} \\
\text{subject to} \quad & \sum_{i \in \mathcal{F}} x_{ij} = 1 \qquad \forall j \in \mathcal{C} \\
& x_{ij} \leq y_i \qquad \forall i \in \mathcal{F}, j \in \mathcal{C} \\
& \sum_{i \in \mathcal{F}} w_i y_i \leq B \\
& 0 \leq x_{ij}, y_i \leq 1 \ \forall i \in \mathcal{F}, j \in \mathcal{C}
\end{aligned}
$$

In this LP, x_{ij} and y_i are indicator variables for the event client j is connected to facility i and facility i is open, respectively. The first constraint guarantees that each client is connected to some facility. The second constraint says that client j can only connect to facility i if it is open. The third one is the knapsack constraint.

In this paper, given a KM instance $\mathcal{I} = (B, \mathcal{F}, \mathcal{C}, c, w)$, let $\text{OPT}_\mathcal{I}$ and OPT_f be the cost of an optimal integral solution and the optimal value of the LP relaxation, respectively. Suppose $\mathcal{S} \subseteq \mathcal{F}$ is a solution to $\mathcal{I}$, let $\text{cost}_\mathcal{I}(\mathcal{S})$ denote cost of $\mathcal{S}$. Let (x, y) denote the optimal (fractional) solution of the LP relaxation. Let $C_j := \sum_{i \in \mathcal{F}} c_{ij} x_{ij}$ be the fractional connection cost of j. Given $\mathcal{S} \subseteq \mathcal{F}$ and a vector $v \in \mathbb{R}^{|\mathcal{F}|}$, let $v(\mathcal{S}) := \sum_{i \in \mathcal{S}} v_i$. From now on, let us fix any optimal integral solution of the instance for the analysis.

2 An improved approximation algorithm for Knapsack Median

2.1 Kumar's bound

The main technical difficulty of KM is related to the unbounded integrality gap of the LP relaxation. It is known that this gap remains unbounded even when we strengthen the LP with knapsack cover inequalities [6]. All previous constant-factor approximation algorithms for KM rely on Kumar's bound from [7] to get around the gap. Specifically, Kumar's bound is useful to bound the connection cost of a group of clients via some <u>cluster center</u> in terms of $\text{OPT}_\mathcal{I}$ instead of OPT_f. We now review this bound, and will improve it later.

Lemma 1 *For each client j, we can compute (in polynomial time) an upper-bound U_j on the connection cost of j in the optimal integral solution (i.e. $cost(j) \leq U_j$) such that*

$$\sum_{j' \in \mathcal{C}} \max\{0, U_j - c_{jj'}\} \leq \mathrm{OPT}_{\mathcal{I}}.$$

Proof We first guess $\mathrm{OPT}_{\mathcal{I}}$ by enumerating the powers of $(1 + \epsilon)$ for some small constant $\epsilon > 0$. (We lose a factor of $(1 + \epsilon)$ in the approximation ratio and a factor of $O(\log n/\epsilon)$ in the runtime.) Now fix any optimal solution and assume that j connects to i and j' connects to i'. Then, by triangle inequality,

$$\mathrm{cost}\,(j) = c_{ij} \leq c_{i'j} \leq c_{jj'} + c_{i'j'} = c_{jj'} + \mathrm{cost}\,(j'),$$

or equivalently,

$$\mathrm{cost}\,(j') \geq \mathrm{cost}\,(j) - c_{jj'}.$$

Taking the sum over all $j' \neq j$, we have

$$\mathrm{OPT}_{\mathcal{I}} \geq \sum_{j' \neq j} \max\{0, \mathrm{cost}\,(j) - c_{jj'}\}.$$

Then we can simply take U_j such that

$$\mathrm{OPT}_{\mathcal{I}} = \sum_{j' \in \mathcal{C}} \max\{0, U_j - c_{jj'}\}.$$

(Observe that the RHS is a linear function of U_j). $\qquad\qquad\square$

We can slightly strengthen the LP relaxation by adding the constraints: $x_{ij} = 0$ for all $c_{ij} > U_j$. (Unfortunately, the integrality gap is still unbounded after this step.) Thus we may assume that (x, y) satisfies all these constraints.

Lemma 2 (Kumar's bound) *Let $\mathcal{S}$ be a set of clients and $s \in \mathcal{S}$, where $c_{js} \leq \beta C_j$ for all $j \in S$ and some constant $\beta \geq 1$, then*

$$|\mathcal{S}|U_s \leq \mathrm{OPT}_{\mathcal{I}} + \beta \sum_{j \in \mathcal{S}} C_j.$$

Proof

$$|\mathcal{S}|U_s = \sum_{j \in \mathcal{S}} U_s = \sum_{j \in \mathcal{S}}(U_s - c_{js}) + \sum_{j \in \mathcal{S}} c_{js} \leq \mathrm{OPT}_{\mathcal{I}} + \beta \sum_{j \in \mathcal{S}} C_j,$$

where we use the property of U_s from Lemma 1 for the last inequality. $\quad\square$

This bound allows one to bound the cost of clients which rely on the bad facility.

2.2 Sparse Instances

Kumar's bound can only be tight when the connection cost in the optimal solution is highly concentrated around a single client. However, if this were the case, we could guess the client for which this occurs, along with its optimal facility, which would give us a large advantage. On the other hand, if the connection cost is evenly distributed, we can greatly strengthen Kumar's bound. This is the idea behind our definition of sparse instances below.

Let $\text{CBall}(j, r) := \{k \in \mathcal{C} : c_{jk} \leq r\}$ denote the set of clients within radius of r from client j. Let λ_j be the connection cost of j in the optimal integral solution. Also, let $i(j)$ denote the facility serving j in the optimal solution.

Definition 1 Given some constants $0 < \delta, \epsilon < 1$, we say that a knapsack median instance $\mathcal{I} = (B, \mathcal{F}, \mathcal{C}, c, w)$ is (δ, ϵ)-sparse if, for all $j \in \mathcal{C}$,

$$\sum_{k \in \text{CBall}(j, \delta\lambda_j)} (\lambda_j - c_{jk}) \leq \epsilon \text{OPT}_{\mathcal{I}}.$$

We will show that the integrality gap is bounded on these sparse instances. We also give a polynomial-time algorithm to sparsify any knapsack median instance. Moreover, the solution of a sparse instance can be used as a solution of the original instance with only a small loss in the total cost.

Lemma 3 *Given some knapsack median instance $\mathcal{I}_0 = (B, \mathcal{F}, \mathcal{C}_0, c, w)$ and $0 < \delta, \epsilon < 1$, there is an efficient algorithm that outputs $O(n^{2/\epsilon})$ pairs of $(\mathcal{I}, \mathcal{F}')$, where $\mathcal{I} = (B, \mathcal{F}, \mathcal{C}, c, w)$ is a new instance with $\mathcal{C} \subseteq \mathcal{C}_0$, and $\mathcal{F}' \subseteq \mathcal{F}$ is a partial solution of $\mathcal{I}$, such that at least one of these instances is (δ, ϵ)-sparse.*

Proof Fix any optimal integral solution of $\mathcal{I}_0$. Consider the following algorithm that transform $\mathcal{I}_0$ into a sparse instance, *assuming for now that we know its optimal solution*:

- Initially, $\mathcal{C} := \mathcal{C}_0$.
- While the instance $(B, \mathcal{F}, \mathcal{C}, c, w)$ is not sparse, i.e. there exists a "bad" client j such that $\sum_{k \in \text{CBall}(j, \delta\lambda_j)}(\lambda_j - c_{jk}) > \epsilon \text{OPT}_{\mathcal{I}}$, <u>remove</u> all clients in $\text{CBall}(j, \delta\lambda_j)$ from $\mathcal{C}$.

Note that this algorithm will terminate after at most $1/\epsilon$ iterations: for each $k \in \text{CBall}(j, \delta\lambda_j)$ and its serving facility $i(k)$ in the optimal solution, we have $c_{ji(k)} \leq c_{jk} + \lambda_k$, which implies

$$\sum_{k \in \text{CBall}(j, \delta\lambda_j)} \lambda_k \geq \sum_{k \in \text{CBall}(j, \delta\lambda_j)} (c_{ji(k)} - c_{jk}) \geq \sum_{k \in \text{CBall}(j, \delta\lambda_j)} (\lambda_j - c_{jk}) > \epsilon \text{OPT}_{\mathcal{I}},$$

and there can be at most $1/\epsilon$ such balls.

Now, while we do not know which client j is "bad" and which facility i serves client j in the optimal solution, we can still guess these pairs in $O(n^2)$

time in each iteration. Specifically, we will guess the number of iterations that the above algorithm terminates and the pair $(j, i(j))$ in each iteration. There are at most $O(n^{2/\epsilon})$ possible cases and we will generate all of these new instances. Finally, we include all the facilities $i(j)$ during the process into the set $\mathcal{F}'$ of the corresponding instance. $\square$

The following theorem says that if we have an approximate solution to a sparse instance, then its cost on the original instance can be blown up by a small constant factor.

Theorem 1 *Let* $\mathcal{I} = (B, \mathcal{F}, \mathcal{C}, c, w)$ *be a* (δ, ϵ)-*sparse instance obtained from* $\mathcal{I}_0 = (B, \mathcal{F}, \mathcal{C}_0, c, w)$ *(by the procedure in the proof of Lemma 3) and* $\mathcal{F}'$ *be the corresponding partial solution. If* $\mathcal{S} \supseteq \mathcal{F}'$ *is any approximate solution to* $\mathcal{I}$ *(including those open facilities in* $\mathcal{F}'$*) such that*

$$cost_{\mathcal{I}}(\mathcal{S}) \le \alpha \mathrm{OPT}_{\mathcal{I}},$$

then

$$cost_{\mathcal{I}_0}(\mathcal{S}) \le \max\left\{\frac{1+\delta}{1-\delta}, \alpha\right\} \mathrm{OPT}_{\mathcal{I}_0}.$$

Note that our notion of sparsity differs from that of Li and Svensson in several ways. It is client-centric, and removes clients instead of facilities from the instance. On the negative side, removed clients' costs blow up by $\frac{1+\delta}{1-\delta}$, so our final approximation cannot guarantee better.

Proof (Theorem 1) For any $k \in \mathcal{C}_0 \setminus \mathcal{C}$, let $\mathrm{CBall}(j, \delta\lambda_j)$ be the ball containing k that was removed from $\mathcal{C}_0$ in the preprocessing phase in Lemma 3. Recall that $i(j)$ is the facility serving j in the optimal solution. We have

$$\lambda_k \ge \lambda_j - c_{jk} \ge (1-\delta)\lambda_j,$$

which implies,

$$c_{ki(j)} \le c_{jk} + \lambda_j \le (1+\delta)\lambda_j \le \frac{1+\delta}{1-\delta}\lambda_k.$$

Then, by connecting all $k \in \mathcal{C}_0 \setminus \mathcal{C}$ to the corresponding facility $i(j)$ (which is guaranteed to be open because $i(j) \in \mathcal{F}'$), we get

$$
\begin{aligned}
\mathrm{cost}_{\mathcal{I}_0}(\mathcal{S}) &= \sum_{k \in \mathcal{C}_0 \setminus \mathcal{C}} \mathrm{cost}(k) + \sum_{k \in \mathcal{C}} \mathrm{cost}(k) \\
&\le \frac{1+\delta}{1-\delta} \sum_{k \in \mathcal{C}_0 \setminus \mathcal{C}} \lambda_k + \alpha \mathrm{OPT}_{\mathcal{I}} \\
&\le \frac{1+\delta}{1-\delta} \sum_{k \in \mathcal{C}_0 \setminus \mathcal{C}} \lambda_k + \alpha \sum_{k \in \mathcal{C}} \lambda_k \\
&\le \max\left\{\frac{1+\delta}{1-\delta}, \alpha\right\} \mathrm{OPT}_{\mathcal{I}_0}.
\end{aligned}
$$

$\square$

From now on, assume that we are given some arbitrary knapsack median instance $\mathcal{I}_0 = (B, \mathcal{F}, \mathcal{C}_0, c, w)$. We will transform $\mathcal{I}_0$ into a (δ, ϵ)-sparse instance $\mathcal{I}$ and use Theorem 1 to bound the real cost at the end.

2.3 Improving Kumar's bound and modifying the LP relaxation

We will show how to improve Kumar's bound in sparse instances. Recall that, for all $j \in \mathcal{C}$, we have

$$\sum_{k \in \mathrm{CBall}(j, \delta\lambda_j)} (\lambda_j - c_{jk}) \leq \epsilon\mathrm{OPT}_{\mathcal{I}}.$$

Then, as before, we can guess $\mathrm{OPT}_{\mathcal{I}}$ and take the maximum U_j such that

$$\sum_{k \in \mathrm{CBall}(j, \delta U_j)} (U_j - c_{jk}) \leq \epsilon\mathrm{OPT}_{\mathcal{I}}.$$

(Observe that the LHS is an increasing function of U_j.) Now the constraints $x_{ij} = 0$ for all $i \in \mathcal{F}, j \in \mathcal{C} : c_{ij} > U_j$ are valid and we can add these into the LP. We also add the following constraints: $y_i = 1$ for all facilities $i \in \mathcal{F}'$. From now on, assume that (x, y) is an optimal solution of this new LP, satisfying all the mentioned constraints.

Lemma 4 *Let s be any client in sparse instance $\mathcal{I}$ and $\mathcal{S}$ be a set of clients such that $c_{js} \leq \beta C_j$ for all $j \in \mathcal{S}$ and some constant $\beta \geq 1$. Then*

$$|\mathcal{S}|U_s \leq \epsilon\mathrm{OPT}_{\mathcal{I}} + \frac{\beta}{\delta} \sum_{j \in \mathcal{S}} C_j.$$

Proof Consider the following two cases.

- For clients $j \in \mathcal{S}' = \mathcal{S} \cap \mathrm{CBall}(s, \delta U_s)$, by definition of sparsity, we have

$$|\mathcal{S}'|U_s = \sum_{j \in \mathcal{S}'} (U_s - c_{js}) + \sum_{j \in \mathcal{S}'} c_{js}$$
$$\leq \epsilon\mathrm{OPT}_{\mathcal{I}} + \beta \sum_{j \in \mathcal{S}'} C_j.$$

- For clients $j \in \mathcal{S}'' = \mathcal{S} \setminus \mathrm{CBall}(s, \delta U_s)$, we have $\beta C_j \geq c_{js} \geq \delta U_s$ and we get an alternative bound $U_s \leq \frac{\beta}{\delta} C_j$. Thus,

$$|\mathcal{S}''|U_s = \sum_{j \in \mathcal{S}''} U_s \leq \sum_{j \in \mathcal{S}''} \frac{\beta}{\delta} C_j.$$

The lemma follows by taking the sum of these two cases. $\square$

2.4 Filtering Phase

We will apply the standard filtering method for facility-location problems (see [3,10]). Basically, we choose a subset $\mathcal{C}' \subseteq \mathcal{C}$ such that clients in $\mathcal{C}'$ are <u>far</u> from each other. After assigning each facility to the closest client in $\mathcal{C}'$, it is possible to lower-bound the opening volume of each cluster. Each client in $\mathcal{C}'$ is called a cluster center.

Filtering algorithm: Initialize $\mathcal{C}' := \mathcal{C}$. For each client $j \in \mathcal{C}'$ in increasing order of C_j, we remove all other clients j' such that $c_{jj'} \le 4C_{j'} = 4\max\{C_{j'}, C_j\}$ from $\mathcal{C}'$.

For each $j \in \mathcal{C}'$, define $F_j = \{i \in \mathcal{F} : c_{ij} = \min_{k \in \mathcal{C}'} c_{ik}\}$, breaking ties arbitrarily. Let $F'_j = \{i \in F_j : c_{ij} \le 2C_j\}$ and $\gamma_j = \min_{i \notin F_j} c_{ij}$. Then define $G_j = \{i \in F_j : c_{ij} \le \gamma_j\}$. We also reassign $y_i := x_{ij}$ for $i \in G_j$ and $y_i := 0$ otherwise. For $j \in \mathcal{C}'$, let M_j be the set containing j and all clients removed by j in the filtering process.

We note that the solution (x, y) may not be feasible to the LP anymore after the reassignment step. For the rest of the paper, we will focus on rounding y into an integral vector. One important property is that the knapsack constraint still holds. In other words, the new sum $\sum_{i \in \mathcal{F}} w_i y_i$ is still at most the budget B. This is due to the fact that $x_{ij} \le y_i$. The opening variables only decrease after this step; and hence, the knapsack constraint will be preserved.

Lemma 5 *We have the following properties:*

- *All sets G_j are disjoint,*
- $1/2 \le y(F'_j)$ *and* $y(G_j) \le 1$ *for all* $j \in \mathcal{C}'$.
- $F'_j \subseteq G_j$ *for all* $j \in \mathcal{C}'$.

Proof For the first claim, observe that all F_j's are disjoint and $G_j \subseteq F_j$ by definition. Also, if $\sum_{i \in F'_j} y_i = \sum_{i \in F'_j} x_{ij} < 1/2$, then $\sum_{i \in \mathcal{F} \setminus F'_j} x_{ij} > 1/2$. Since the radius of F'_j is $2C_j$, this means that $C_j > (1/2)(2C_j) = C_j$, which is a contradiction. Since we reassign $y_i := x_{ij}$ for all $i \in G_j$, the volume $y(G_j)$ is now at most 1. Finally, we have $2C_j \le \gamma_j$. Otherwise, let $i \notin F_j$ be the facility such that $\gamma_j = c_{ij}$. Observe that facility i is claimed by another cluster center, say j', because $c_{ij'} \le c_{ij} \le 2C_j$. This implies that $c_{jj'} \le c_{ij} + c_{ij'} \le 4C_j$, which is a contradiction. $\qquad\square$

It is clear that for all $j, j' \in \mathcal{C}'$, $c_{jj'} \ge 4\max\{C_{j'}, C_j\}$. Moreover, for each $j \in \mathcal{C} \setminus \mathcal{C}'$, we can find $j' \in \mathcal{C}'$, where j' causes the removal of j, or, in other words, $C_{j'} \le C_j$ and $c_{jj'} \le 4C_j$. Assuming that we have a solution $\mathcal{S}$ for the instance $\mathcal{I}' = (B, \mathcal{F}, \mathcal{C}', c, w)$ where each client j in $\mathcal{C}'$ has demand $d_j = |M_j|$ (i.e. there are $|M_j|$ copies of j), we can transform it into a solution for $\mathcal{I}$ as follows. Each client $j \in \mathcal{C} \setminus \mathcal{C}'$ will be served by the facility of j' that removed

j. Then $\operatorname{cost}(j) = c_{jj'} + \operatorname{cost}(j') \le \operatorname{cost}(j') + 4C_j$. Therefore,

$$
\begin{aligned}
\operatorname{cost}_{\mathcal{I}}(\mathcal{S}) &= \sum_{j \in \mathcal{C}'} \operatorname{cost}(j) + \sum_{j \in \mathcal{C} \setminus \mathcal{C}'} \operatorname{cost}(j) \\
&\le \sum_{j \in \mathcal{C}'} \operatorname{cost}(j) + \sum_{j \in \mathcal{C} \setminus \mathcal{C}'} \left(\operatorname{cost}(j'(j)) + 4C_j \right) \\
&\le \operatorname{cost}_{\mathcal{I}'}(\mathcal{S}) + 4\mathrm{OPT}_f.
\end{aligned}
$$

where, in the second line, $j'(j)$ is the center in $\mathcal{C}'$ that removed j.

2.5 A basic $(23.09 + \epsilon)$-approximation algorithm

In this section, we describe a simple randomized $(23.09 + \epsilon)$-approximation algorithm. In the next section, we will derandomize it and give more insights to further improve the approximation ratio to $17.46 + \epsilon$.

High-level ideas: We reuse Swamy's idea from [10] to first obtain an almost half integral solution $\hat{y}$. This solution $\hat{y}$ has a very nice structure. For example, each client j only (fractionally) connects to at most 2 facilities, and there is at least a half-opened facility in each G_j. We shall refer to this set of 2 facilities as a bundle. In [10], the author applies a standard clustering process to get disjoint bundles and round $\hat{y}$ by opening at least one facility per bundle. The drawback of this method is that we have to pay extra cost for bundles removed in the clustering step. In fact, it is possible to open at least one facility per bundle without filtering out any bundle. The idea here is inspired by the work of Charikar et. al [2]. In addition, instead of picking $\hat{y}$ deterministically, sampling such a half integral extreme point will be very helpful for the analysis.

We consider the following polytope.

$$
\mathcal{P} = \{ v \in [0,1]^{|\mathcal{F}|} : v(F'_j) \ge 1/2, \ v(G_j) \le 1, \ \forall j \in \mathcal{C}'; \ \sum_{i \in \mathcal{F}} w_i v_i \le B \}.
$$

Lemma 6 ([10]) *Any extreme point of $\mathcal{P}$ is almost half-integral: there exists at most 1 cluster center $s \in \mathcal{C}'$ such that G_s contains variables $\notin \{0, \frac{1}{2}, 1\}$. We call s a fractional client.*

Notice by Lemma 5 that $y \in \mathcal{P}$. By Carathéodory's theorem, y is a convex combination of at most $t = |\mathcal{F}| + 1$ extreme points of $\mathcal{P}$. Moreover, there is an efficient algorithm based on the ellipsoid method to find such a decomposition (e.g., see [9]). We apply this algorithm to get extreme points $y^{(1)}, y^{(2)}, \ldots, y^{(t)} \in \mathcal{P}$ and coefficients $0 \le p_1, \ldots, p_t \le 1, \sum_{i=1}^{t} p_i = 1$, such that

$$
y = p_1 y^{(1)} + p_2 y^{(2)} + \ldots + p_t y^{(t)}.
$$

This representation defines a distribution on t extreme points of $\mathcal{P}$. Let $Y \in [0,1]^{\mathcal{F}}$ be a random vector where $\Pr[Y = y^{(i)}] = p_i$ for $i = 1, \ldots, t$.

Observe that Y is <u>almost</u> half-integral. Let s be the fractional client in Y. (We assume that s exists; otherwise, the cost will only be smaller.)

Defining primary and secondary facilities: For each $j \in \mathcal{C}'$,

- If $j \neq s$, let $i_1(j)$ be any half-integral facility in F'_j (i.e. $Y_{i_1(j)} = 1/2$; such a facility exists because $Y(F'_j) \geq 1/2$). Else ($j = s$), let $i_1(j)$ be the smallest-weight facility in F'_j with $Y_{i_1(j)} > 0$.
- If $Y(i_1(j)) = 1$, let $i_2(j) = i_1(j)$.
- If $Y(G_j) < 1$, then let $\sigma(j)$ be the nearest client to j in $\mathcal{C}'$. Define $i_2(j) = i_1(\sigma(j))$.
- If $Y(G_j) = 1$, then
    - If $j \neq s$, let $i_2(j)$ be the other half-integral facility in G_j.
    - Else ($j = s$), let $i_2(j)$ be the smallest-weight facility in G_j with $Y_{i_2(j)} > 0$. If there are ties and $i_1(j)$ is among these facilities then we let $i_2(j) = i_1(j)$.
- We call $i_1(j), i_2(j)$ the primary facility and the secondary facility of j, respectively.

Constructing the neighborhood graph: Initially, construct the directed graph $\mathcal{G}$ on clients in $\mathcal{C}'$ such that there is an edge $j \to \sigma(j)$ for each $j \in \mathcal{C}'$: $Y(G_j) < 1$. Note that all vertices in $\mathcal{G}$ have outdegree ≤ 1. If $Y(G_j) = 1$, then vertex j has no outgoing edge. In this case, we replace j by the edge $i_1(j) \to i_2(j)$, instead. Finally, we relabel all other nodes in $\mathcal{G}$ by its primary facility. Now we can think of each client $j \in \mathcal{C}'$ as an edge from $i_1(j)$ to $i_2(j)$ in $\mathcal{G}$.

Lemma 7 *Without loss of generality, we can assume that all cycles of $\mathcal{G}$ (if any) are of size 2. This means that $\mathcal{G}$ is bipartite.*

Proof Since the maximum outdegree is equal to 1, each (weakly) connected component of $\mathcal{G}$ has at most 1 cycle. Consider any cycle $j \to \sigma(j) \to \sigma^2(j) \to \ldots \to \sigma^k(j) \to j$. Then it is easy to see that $c_{j\sigma(j)} = c_{\sigma^k(j)j}$. The argument holds for any j in the cycle, and all edges on the cycle have the same length. Then we can simply redefine $\sigma(\sigma^k(j)) := \sigma^{k-1}(j)$ and get a cycle of size 2 instead. We can also change the secondary of the client corresponding to the edge $(\sigma^k(j), j)$ into $\sigma^{k-1}(j)$ because they are both at the same distance from it. □

We are now ready to describe the main algorithm.

Theorem 2 *Algorithm 2 returns a feasible solution $\mathcal{S}$ where*

$$\mathrm{E}[\,cost_{\mathcal{I}_0}(\mathcal{S})] \leq \max\left\{\frac{1+\delta}{1-\delta}, 10 + 12/\delta + 3\epsilon\right\} \mathrm{OPT}_{\mathcal{I}_0}.$$

In particular, the approximation ratio is at most $(23.087 + 3\epsilon)$ when setting $\delta := 0.916966$.

Algorithm 1 ROUND(Y)

1: Construct the neighborhood graph $\mathcal{G}$ based on Y
2: Let C_1, C_2 be independent sets which partition $\mathcal{G}$.
3: Let W_1, W_2 be the total weight of the facilities in C_1, C_2 respectively.
4: **if** $W_1 \leq W_2$ **then**
5:     **return** C_1
6: **else**
7:     **return** C_2

Algorithm 2 BASICALGORITHM$(\delta, \epsilon, \mathcal{I}_0)$

1: Generate $O(n^{2/\epsilon})$ pairs $(\mathcal{I}, \mathcal{F}')$ using the algorithm in the proof of Lemma 3.
2: $\mathcal{S} \leftarrow \emptyset$
3: **for** each pair $(\mathcal{I}, \mathcal{F}')$ **do**
4:     Let (x, y) be the optimal solution of the modified LP relaxation in Section 2.3.
5:     Apply the filtering algorithm to get $\mathcal{I}'$.
6:     Use $\mathcal{F}, \mathcal{C}'$ to define the polytope $\mathcal{P}$.
7:     Sample a random extreme point Y of $\mathcal{P}$ as described above.
8:     Let $\mathcal{S}' \leftarrow$ROUND(Y)
9:     If $\mathcal{S}'$ is feasible and its cost is smaller than the cost of $\mathcal{S}$ then $\mathcal{S} \leftarrow \mathcal{S}'$.
10: **return** $\mathcal{S}$

Proof Assume $\mathcal{I}$ is the sparse instance obtained from $\mathcal{I}_0$. We will give a proof of feasibility and a cost analysis. Recall that s is the center where we may have fractional values Y_i with $i \in G_s$.

Feasibility:

– For all centers $j \in \mathcal{C}'$ with $Y(G_j) < 1$, we have

$$w_{i_1(j)} \leq 2 \sum_{i \in G_j} Y_i w_i.$$

Note that this is true for $j \neq s$ because $Y_{i_1(j)} = 1/2$. Otherwise, $j = s$, by definition, $w_{i_1(j)}$ is the smallest weight in the set F'_s which has volume at least $1/2$. Thus, $w_{i_1(j)} \leq 2 \sum_{i \in F'_s} Y_i w_i \leq 2 \sum_{i \in G_j} Y_i w_i$.

– For all centers $j \in \mathcal{C}'$ with $Y(G_j) = 1$, we have

$$w_{i_1(j)} + w_{i_2(j)} \leq 2 \sum_{i \in G_j} Y_i w_i.$$

The equality happens when $j \neq s$. Otherwise, $j = s$, we consider the following 2 cases

– If $i_1(s) = i_2(s)$ the inequality follows because $w_{i_1(j)} = w_{i_2(j)} \leq \sum_{i \in G_j} Y_i w_i$.
– Else, we have $i_2(s) \in G_j \setminus F'_j$ by definition of $i_2(s)$. Since $w_{i_1(s)} \geq w_{i_2(s)}$ and $Y(F'_s) \geq 1/2$,

$$\frac{1}{2} w_{i_1(s)} + \frac{1}{2} w_{i_2(s)} \leq Y(F'_s) w_{i_1(s)} + (1 - Y(F'_s)) w_{i_2(s)}$$

$$\leq \sum_{i \in F'_j} Y_i w_i + \sum_{i \in G_j \setminus F'_j} Y_i w_i = \sum_{i \in G_j} Y_i w_i.$$

Recall that each center $j \in \mathcal{C}'$ is accounted for either one vertex $i_1(j)$ of $\mathcal{G}$ if $Y(G_j) < 1$ or two vertices $i_1(j), i_2(j)$ of $\mathcal{G}$ if $Y(G_j) = 1$). Thus, the total weight of all vertices in $\mathcal{G}$ is at most

$$2 \sum_{j \in \mathcal{C}'} \sum_{i \in G_j} Y_i w_i \leq 2B,$$

where the last inequality follows because $Y \in \mathcal{P}$. It means that either W_1 or W_2 is less than or equal to B, and Algorithm 1 always returns a feasible solution.

Cost analysis:

We show that the expected cost of j can be bounded in terms of γ_j, U_j, and y. For $j \in \mathcal{C}$, let $j'(j)$ denote the cluster center of j and define $j'(j) = j$ if $j \in \mathcal{C}'$. Recall that in the instance $\mathcal{I}' = (B, \mathcal{F}, \mathcal{C}', c)$, each client $j \in \mathcal{C}'$ has demand $d_j = |M_j|$. Notice that

$$\mathrm{OPT}_f = \sum_{j \in \mathcal{C}} C_j \geq \sum_{j \in \mathcal{C}} C_{j'(j)} = \sum_{j \in \mathcal{C}'} d_j C_j$$

$$= \sum_{j \in \mathcal{C}'} d_j \left(\sum_{i \in G_j} x_{ij} c_{ij} + \sum_{i \in \mathcal{F} \setminus G_j} x_{ij} c_{ij} \right)$$

$$\geq \sum_{j \in \mathcal{C}'} d_j \left(\sum_{i \in G_j} y_i c_{ij} + \gamma_j (1 - y(G_j)) \right). \tag{1}$$

The last inequality follows because, for any center j, $\sum_{i \in \mathcal{F}} x_{ij} = 1$, and γ_j is the radius of the ball G_j by definition. Now, for $v \in [0,1]^{\mathcal{F}}$, we define

$$B_j(v) := d_j \left(\sum_{i \in G_j} v_i c_{ij} + \gamma_j (1 - v(G_j)) \right).$$

Let $K(v) = \sum_{j \in \mathcal{C}'} B_j(v)$. Recall that $\mathrm{E}[Y_i] = y_i$ for all $i \in \mathcal{F}$. By (1) and linearity of expectation, we have

$$\mathrm{E}[K(Y)] = K(y) \leq \mathrm{OPT}_f.$$

Also note that

$$\sum_{j \in \mathcal{C}'} \mathrm{E}[B_j(Y)] = \sum_{j \in \mathcal{C}'} B_j(y) \leq \sum_{j \in \mathcal{C}'} d_j C_j \leq \sum_{j \in \mathcal{C}'} \sum_{k \in M_j} C_k = \sum_{j \in \mathcal{C}} C_j.$$

Next, we will analyze $\mathrm{cost}_{\mathcal{I}'}(\mathcal{S})$. To this end, we shall bound the connection cost of a client j in terms of $B_j(Y)$. Algorithm 1 guarantees that, for each $j \in \mathcal{C}'$, either $i_1(j)$ or $i_2(j)$ is in $\mathcal{S}$. By construction, $c_{i_1(j)j} \leq c_{i_2(j)j}$. In the worst case, we may need to connect j to $i_2(j)$, and hence $\mathrm{cost}(j) \leq d_j c_{i_2(j)j}$ for all client j.

Fix any client j with $Y(G_j) < 1$. Recall that $\gamma_j = \min_{i \notin F_j} c_{ij}$ and $\sigma(j)$ is the closest client to j in $\mathcal{C}'$. Suppose $\gamma_j = c_{i'j}$ where $i' \in F_{j'}$ for some $j' \in \mathcal{C}'$. By definition, $c_{i'j'} \leq \gamma_j$. Then $c_{j\sigma(j)} \leq c_{jj'} \leq c_{i'j} + c_{i'j'} \leq 2\gamma_j$. Also, since $i_1(\sigma(j)) \in F'_{\sigma(j)}$, we have that $c_{\sigma(j)i_1(\sigma(j))} \leq 2C_{\sigma(j)}$. In addition, recall that $4\max\{C_j, C_{\sigma(j)}\} \leq c_{j\sigma(j)} \leq 2\gamma_j$. Thus, $2C_{\sigma(j)} \leq \gamma_j$. Then the following bound holds when $Y(G_j) < 1$:

$$
\begin{aligned}
\text{cost}\,(j) &\leq d_j c_{i_2(j)j} \\
&\leq d_j(c_{j\sigma(j)} + c_{\sigma(j)i_2(j)}) \\
&= d_j(c_{j\sigma(j)} + c_{\sigma(j)i_1(\sigma(j))}) \\
&\leq d_j(2\gamma_j + 2C_{\sigma(j)}) \\
&\leq 3d_j\gamma_j.
\end{aligned}
$$

Consider the following cases.

- If $j \neq s$, then either $Y(G_j) = 1$ or $Y(G_j) = 1/2$.
  - Case $Y(G_j) = 1$: then $Y_{i_1(j)} = Y_{i_2(j)} = 1/2$, we have

$$
\text{cost}\,(j) \leq d_j c_{i_2(j)j} \leq 2d_j \sum_{i \in G_j} Y_i c_{ij} = 2B_j(Y).
$$

  - Case $Y(G_j) = 1/2$: we have

$$
\text{cost}\,(j) \leq 3d_j\gamma_j = 6d_j\gamma_j(1 - Y(G_j)) \leq 6B_j(Y).
$$

- If $j = s$, we cannot bound the cost in terms of $B_j(Y)$. Instead, we shall use Kumar's bound.
  - Case $Y(G_j) = 1$: $i_2(j) \in G_j$. Recall that U_j is the upper-bound on the connection cost of j. Our LP constraints guarantee that $x_{ij} = 0$ for all $c_{ij} > U_j$. Since $Y_{i_2(j)} > 0$, we also have $y_{i_2(j)} > 0$ or $x_{i_2(j)j} > 0$, which implies that $c_{i_2(j)j} \leq U_j$. Thus,

$$
\text{cost}\,(j) \leq d_j c_{i_2(j)j} \leq d_j U_j.
$$

  - Case $Y(G_j) < 1$: then there must exists some facility $i \notin G_j$ such that $x_{ij} > 0$. Since γ_j is the radius of G_j, we have $\gamma_j \leq c_{ij} \leq U_j$; and hence,

$$
\text{cost}\,(j) \leq 3d_j\gamma_j \leq 3d_j U_j.
$$

In either cases, applying the improved Kumar's bound to the cluster M_s where $c_{ks} \leq 4C_k$ for all $k \in M_s$, we get

$$
\begin{aligned}
\text{cost}\,(j) &\leq 3d_j U_j \\
&\leq 3\epsilon\text{OPT}_{\mathcal{I}} + \frac{3 \cdot 4}{\delta} \sum_{k \in M_s} C_k \\
&\leq 3\epsilon\text{OPT}_{\mathcal{I}} + \frac{12}{\delta}\text{OPT}_f.
\end{aligned}
$$

Now, we will bound the facility-opening cost. Notice that, for all facilities $i \in C_1 \cup C_2$ but at most two facilities $i_1(s)$ and $i_2(s)$, we have $Y_i \in \{1/2, 1\}$.

Then,

$$
\begin{aligned}
\mathrm{E}[\operatorname{cost}_{\mathcal{I}'}(\mathcal{S})] &\leq \sum_{j \in \mathcal{C}': j \neq s} 6\,\mathrm{E}[B_j(Y)] + 3\epsilon\mathrm{OPT}_{\mathcal{I}} + (12/\delta)\mathrm{OPT}_f \\
&= 6 \sum_{j \in \mathcal{C}'} B_j(y) + 3\epsilon\mathrm{OPT}_{\mathcal{I}} + (12/\delta)\mathrm{OPT}_f \\
&\leq 6K(y) + 3\epsilon\mathrm{OPT}_{\mathcal{I}} + (12/\delta)\mathrm{OPT}_f \\
&\leq (6 + 12/\delta)\mathrm{OPT}_f + 3\epsilon\mathrm{OPT}_{\mathcal{I}}.
\end{aligned}
$$

Therefore,

$$
\mathrm{E}[\operatorname{cost}_{\mathcal{I}}(\mathcal{S})] \leq \mathrm{E}[\operatorname{cost}_{\mathcal{I}'}(\mathcal{S})] + 4\mathrm{OPT}_f \leq (10 + 12/\delta)\mathrm{OPT}_f + 3\epsilon\mathrm{OPT}_{\mathcal{I}}.
$$

Finally, applying Theorem 1 to $\mathcal{S}$,

$$
\mathrm{E}[\operatorname{cost}_{\mathcal{I}_0}(\mathcal{S})] \leq \max\left\{\frac{1+\delta}{1-\delta}, 10 + 12/\delta + 3\epsilon\right\}\mathrm{OPT}_{\mathcal{I}_0}.
$$

$\square$

2.6 A $(17.46 + \epsilon)$-approximation algorithm via conditioning on the fractional cluster center

Recall that the improved Kumar's bound for the fractional client s is

$$
|M_s|U_s \leq \epsilon\mathrm{OPT}_{\mathcal{I}} + (4/\delta)\sum_{j \in M_s} C_j.
$$

In Theorem 2, we upper-bound the term $\sum_{j \in M_s} C_j$ by OPT_f. However, if this is tight, then the fractional cost of all other clients not in M_s must be zero and we should get an improved ratio.

To formalize this idea, let $u \in \mathcal{C}'$ be the client such that $\sum_{j \in M_u} C_j$ is maximum. Let $\alpha \in [0, 1]$ such that $\sum_{j \in M_u} C_j = \alpha\mathrm{OPT}_f$, then

$$
|M_s|U_s \leq \epsilon\mathrm{OPT}_{\mathcal{I}} + (4/\delta)\alpha\mathrm{OPT}_f. \tag{2}
$$

The following bound follows immediately by replacing the Kumar's bound by (2) in the proof of Theorem 2.

$$
\mathrm{E}[\operatorname{cost}_{\mathcal{I}}(\mathcal{S})] \leq (10 + 12\alpha/\delta + 3\epsilon)\mathrm{OPT}_{\mathcal{I}}. \tag{3}
$$

In fact, this bound is only tight when u happens to be the fractional client after sampling Y. If u is not "fractional", the second term in the RHS of (2) should be at most $(1 - \alpha)\mathrm{OPT}_f$. Indeed, if u is rarely a fractional client, we should obtain a strictly better bound. To this end, let $\mathcal{E}$ be the event that u is the fractional client after the sampling phase. Let $p = \Pr[\mathcal{E}]$. We get the following lemma.

Lemma 8 *Algorithm 2 returns a solution $\mathcal{S}$ with*

$$\mathrm{E}[\,cost_{\mathcal{I}}(\mathcal{S})] \leq (10 + \min\{12\alpha/\delta, (12/\delta)(p\alpha + (1-p)(1-\alpha))\} + 3\epsilon)\mathrm{OPT}_{\mathcal{I}}.$$

Proof We reuse the notations and the connection cost analysis in the proof of Theorem 2. Recall that $\mathcal{E}$ is the event that u is the fractional client. We have

$$\mathrm{E}[\,\mathrm{cost}_{\mathcal{I}'}(\mathcal{S})|\mathcal{E}] \leq 6 \sum_{j \in \mathcal{C}': j \neq u} \mathrm{E}[B_j(Y)|\mathcal{E}] + 3\epsilon\mathrm{OPT}_{\mathcal{I}} + (12\alpha/\delta)\mathrm{OPT}_f.$$

If $\bar{\mathcal{E}}$ happens, assume $s \neq u$ is the fractional one and let $\bar{\mathcal{E}}(s)$ denote this event. Then,

$$\mathrm{E}[\,\mathrm{cost}_{\mathcal{I}'}(\mathcal{S})|\bar{\mathcal{E}}(s)] \leq 6 \sum_{j \in \mathcal{C}': j \neq s} \mathrm{E}[B_j(Y)|\bar{\mathcal{E}}(s)] + 3\epsilon\mathrm{OPT}_{\mathcal{I}} + (12/\delta)(1-\alpha)\mathrm{OPT}_f$$

$$\leq 6 \sum_{j \in \mathcal{C}'} \mathrm{E}[B_j(Y)|\bar{\mathcal{E}}(s)] + 3\epsilon\mathrm{OPT}_{\mathcal{I}} + (12/\delta)(1-\alpha)\mathrm{OPT}_f$$

Therefore,

$$\mathrm{E}[\,\mathrm{cost}_{\mathcal{I}'}(\mathcal{S})|\bar{\mathcal{E}}] \leq 6 \sum_{j \in \mathcal{C}'} \mathrm{E}[B_j(Y)|\bar{\mathcal{E}}] + 3\epsilon\mathrm{OPT}_{\mathcal{I}} + (12/\delta)(1-\alpha)\mathrm{OPT}_f.$$

Also, $(1-p)\,\mathrm{E}[B_u(Y)|\bar{\mathcal{E}}] \leq \mathrm{E}[B_u(Y)]$ because $B_u(Y)$ is always non-negative. The total expected cost can be bounded as follows.

$$\mathrm{E}[\,\mathrm{cost}_{\mathcal{I}'}(\mathcal{S})] = p\,\mathrm{E}[cost_{\mathcal{I}'}(\mathcal{S})|\mathcal{E}] + (1-p)\,\mathrm{E}[cost_{\mathcal{I}'}(\mathcal{S})|\bar{\mathcal{E}}]$$

$$\leq 6 \sum_{j \in \mathcal{C}': j \neq u} \mathrm{E}[B_j(Y)] + 3\epsilon\mathrm{OPT}_{\mathcal{I}}$$

$$+ (12/\delta)(p\alpha + (1-p)(1-\alpha))\mathrm{OPT}_f + 6(1-p)\,\mathrm{E}[B_u(Y)|\bar{\mathcal{E}}]$$

$$\leq 6 \sum_{j \in \mathcal{C}'} \mathrm{E}[B_j(Y)] + 3\epsilon\mathrm{OPT}_{\mathcal{I}} + (12/\delta)(p\alpha + (1-p)(1-\alpha))\mathrm{OPT}_f.$$

$$\leq 6K(y) + (3\epsilon + (12/\delta)(p\alpha + (1-p)(1-\alpha)))\mathrm{OPT}_{\mathcal{I}}$$

$$\leq (6 + 3\epsilon + (12/\delta)(p\alpha + (1-p)(1-\alpha)))\mathrm{OPT}_{\mathcal{I}}. \tag{4}$$

The lemma follows due to (3), (4), and the fact that $\mathrm{E}[\,\mathrm{cost}_{\mathcal{I}}(\mathcal{S})] \leq \mathrm{E}[\,\mathrm{cost}_{\mathcal{I}'}(\mathcal{S})] + 4\mathrm{OPT}_f$. $\qquad\square$

Finally, conditioning on the event $\mathcal{E}$, we are able to combine certain terms and get the following improved bound.

Lemma 9 *Algorithm 2 returns a solution $\mathcal{S}$ with*

$$\mathrm{E}[\,cost_{\mathcal{I}}(\mathcal{S})|\mathcal{E}] \leq (\max\{6/p, 12/\delta\} + 4 + 3\epsilon)\mathrm{OPT}_{\mathcal{I}}.$$

Proof Again, since $B_j(Y) \geq 0$ for all $j \in \mathcal{C}'$ and all Y, we have $\mathrm{E}[B_j(Y)|\mathcal{E}] \leq \mathrm{E}[B_j(Y)]/p$. Also, recall that $\mathrm{E}[B_j(Y)] = B_j(y) \leq d_j C_j \leq \sum_{k \in M_j} C_k$ for any $j \in \mathcal{C}'$. Therefore,

$$
\begin{aligned}
\mathrm{E}[\,\mathrm{cost}_{\mathcal{I}'}(\mathcal{S})|\mathcal{E}] &\leq 6 \sum_{j \in \mathcal{C}': j \neq u} \mathrm{E}[B_j(Y)|\mathcal{E}] + 3\epsilon \mathrm{OPT}_{\mathcal{I}} + (12/\delta) \sum_{j \in M_u} C_j \\
&\leq (6/p) \sum_{j \in \mathcal{C}': j \neq u} \mathrm{E}[B_j(Y)] + 3\epsilon \mathrm{OPT}_{\mathcal{I}} + (12/\delta) \sum_{j \in M_u} C_j \\
&\leq (6/p) \sum_{j \in \mathcal{C}: j \notin M_u} C_j + 3\epsilon \mathrm{OPT}_{\mathcal{I}} + (12/\delta) \sum_{j \in M_u} C_j \\
&\leq \max\{6/p, 12/\delta\} \sum_{j \in \mathcal{C}} C_j + 3\epsilon \mathrm{OPT}_{\mathcal{I}} \\
&\leq \max\{6/p, 12/\delta\} \mathrm{OPT}_f + 3\epsilon \mathrm{OPT}_{\mathcal{I}} \\
&\leq (\max\{6/p, 12/\delta\} + 3\epsilon) \mathrm{OPT}_{\mathcal{I}}.
\end{aligned}
$$

The lemma follows since $\mathrm{E}[\,\mathrm{cost}_{\mathcal{I}}(\mathcal{S})|\mathcal{E}] \leq \mathrm{E}[\,\mathrm{cost}_{\mathcal{I}'}(\mathcal{S})|\mathcal{E}] + 4\mathrm{OPT}_f$. □

Now we have all the required ingredients to get an improved approximation ratio. Algorithm 3 is a derandomized version of Algorithm 2.

Algorithm 3 DETERMINISTICALGORITHM(δ, ϵ, $\mathcal{I}_0$)

1: Generate $O(n^{2/\epsilon})$ pairs $(\mathcal{I}, \mathcal{F}')$ using the algorithm in the proof of Lemma 3.
2: $\mathcal{S} \leftarrow \emptyset$
3: **for** each pair $(\mathcal{I}, \mathcal{F}')$ **do**
4:     Let (x, y) be the optimal solution of the modified LP relaxation in Section 2.3.
5:     Apply the filtering algorithm to get $\mathcal{I}'$.
6:     Use $\mathcal{F}, \mathcal{C}'$ to define the polytope $\mathcal{P}$.
7:     Decompose y into a convex combination of extreme points $y^{(1)}, y^{(2)}, \ldots, y^{(t)}$ of $\mathcal{P}$.
8:     **for** each $Y \in \{y^{(1)}, y^{(2)}, \ldots, y^{(t)}\}$ **do**
9:         Let $\mathcal{S}' \leftarrow$ ROUND(Y)
10:         If $\mathcal{S}'$ is feasible and its cost is smaller than the cost of $\mathcal{S}$ then $\mathcal{S} \leftarrow \mathcal{S}'$.
11: **return** $\mathcal{S}$

Theorem 3 *Algorithm 3 returns a feasible solution $\mathcal{S}$ where*

$$
\mathrm{cost}_{\mathcal{I}_0}(\mathcal{S}) \leq (17.46 + 3\epsilon) \mathrm{OPT}_{\mathcal{I}_0},
$$

when setting $\delta = 0.891647$.

Proof Again, suppose $\mathcal{I}$ is a sparse instance obtained from $\mathcal{I}_0$. Recall that $p = \Pr[\mathcal{E}]$ is the probability that u, the cluster center with maximum fractional cost $\sum_{j \in M_u} C_j = \alpha \mathrm{OPT}_f$, is fractional. Consider the following cases:

– Case $p \leq 1/2$: By Lemma 8 and the fact that Algorithm 3 always returns a solution $\mathcal{S}$ from the same distribution with minimum cost, we have

$$
\mathrm{cost}_{\mathcal{I}}(\mathcal{S}) \leq (10 + \min\{12\alpha/\delta, (12/\delta)(p\alpha + (1-p)(1-\alpha))\} + 3\epsilon) \mathrm{OPT}_{\mathcal{I}}.
$$

By Theorem 1, the approximation ratio is at most

$$\max\left\{\frac{1+\delta}{1-\delta}, 10+3\epsilon+\min\{12\alpha/\delta, (12/\delta)(p\alpha+(1-p)(1-\alpha))\}\right\}.$$

– If $\alpha \leq 1/2$, the ratio is at most $\max\left\{\frac{1+\delta}{1-\delta}, 10+3\epsilon+6/\delta\right\}$.
– If $\alpha \geq 1/2$, we have

$$(12/\delta)(p\alpha+(1-p)(1-\alpha)) = (12/\delta)(p(2\alpha-1)-\alpha+1) \leq 6/\delta.$$

   Again, the ratio is at most $\max\left\{\frac{1+\delta}{1-\delta}, 10+3\epsilon+6/\delta\right\}$.
– Case $p \geq 1/2$: Observe that the event $\mathcal{E}$ does happen for some point in the for loop at lines 8, 9, and 10. By Lemma 9 and the fact that $1+2/p = 3 < 12/\delta$, we have

$$\text{cost}_{\mathcal{I}}(\mathcal{S}) \leq (\max\{6/p, 12/\delta\}+4+3\epsilon)\text{OPT}_{\mathcal{I}} = (12/\delta+4+3\epsilon)\text{OPT}_{\mathcal{I}}.$$

By Theorem 1, the approximation ratio is bounded by $\max\left\{\frac{1+\delta}{1-\delta}, \frac{12}{\delta}+3\epsilon+4\right\}$.

In all cases, the approximation ratio is at most

$$\max\left\{\frac{1+\delta}{1-\delta}, 12/\delta+3\epsilon+4, 10+3\epsilon+6/\delta\right\} \leq 17.4582+3\epsilon,$$

when $\delta = 0.89167$.   □

Note that in [10], Swamy considered a slightly more general version of KM where each facility also has an opening cost. It can be shown that Theorem 3 also extends to this variant.

3 A bi-factor 3.05-approximation algorithm for Knapsack median

In this section, we develop a bi-factor approximation algorithm for KM that outputs a pseudo-solution of cost at most $3.05\text{OPT}_{\mathcal{I}}$ and of weight bounded by $(1+\epsilon)B$. This is a substantial improvement upon the previous comparable result, which achieved a factor of $16+\epsilon$ and violated the budget additively by the largest weight $w_{\max}$ of a facility. It is not hard to observe that one can also use Swamy's algorithm [10] to obtain an 8-approximation that opens a constant number of extra facilities (exceeding the budget B). Our algorithm works for the original problem formulation of KM where all facility costs are zero. Our algorithm is inspired by a recent algorithm of Li and Svensson [8] for the k-median problem, which beat the long standing best bound of $3+\epsilon$. The overall approach consists in computing a so-called bi-point solution, which is a convex combination $a\mathcal{F}_1 + b\mathcal{F}_2$ of two integral pseudo solutions $\mathcal{F}_1$ and $\mathcal{F}_2$ for appropriate factors $a, b \geq 0$ with $a+b = 1$, and then rounding this bi-point solution to an integral one.

Depending on the value of a, Li and Svensson apply three different bi-point rounding procedures. We extend two of them to the case of KM. The rounding procedures of Li and Svensson have the inherent property of opening $k + c$ facilities where c is a constant. Li and Svensson find a way to preprocess the instance such that any pseudo approximation algorithm for k-median that opens $k + c$ facilities can be turned into a (proper) approximation algorithm by paying only an additional ϵ in the approximation ratio. We did not find a way to prove a similar result also for KM and therefore our algorithms violate the facility budget by a factor of $1 + \epsilon$.

3.1 Pruning the instance

The bi-factor approximation algorithm that we will describe in Section 3.2 has the following property. It outputs a (possibly infeasible) pseudo-solution of cost at most $\alpha \mathrm{OPT}_\mathcal{I}$ such that the budget B is respected when we remove the two heaviest facilities from this solution. This can be combined with a simple reduction to the case where the weight of any facility is at most ϵB. This ensures that our approximate solution violates the budget by a factor at most $1 + 2\epsilon$ while maintaining the approximation factor α.

Lemma 10 *Let $\mathcal{I} = (B, \mathcal{F}, \mathcal{C}, c, w)$ be any KM instance. Assume there exists an algorithm A that computes for instance $\mathcal{I}$ a solution that consists of a feasible solution and two additional facilities, and that has cost at most $\alpha \mathrm{OPT}_\mathcal{I}$. Then there exists for any $\epsilon > 0$ a bi-factor approximation algorithm A' which computes a solution of weight $(1 + \epsilon)B$ and of cost at most $\alpha \mathrm{OPT}_\mathcal{I}$.*

Proof Let $\mathcal{I} = (B, \mathcal{F}, \mathcal{C}, c, w)$ be an instance of knapsack median, let $\mathcal{F}_\epsilon \subseteq \mathcal{F}$ be the set of facilities whose weight exceeds ϵB and let $\mathcal{S}$ be some fixed optimum solution. Note that any feasible solution can have no more than $1/\epsilon$ many facilities in $\mathcal{F}_\epsilon$.

This allows us to guess the set $\mathcal{S}_\epsilon := \mathcal{S} \cap \mathcal{F}_\epsilon$ of "heavy" facilities in the optimum solution $\mathcal{S}$. To this end we enumerate all $O(\frac{1}{\epsilon} |\mathcal{F}|^{1/\epsilon})$ many subsets of $\mathcal{F}_\epsilon$ of cardinality at most $1/\epsilon$. At some iteration, we will consider precisely the set $\mathcal{S}_\epsilon$. We modify the instance as follows. The budget is adjusted to $B' := B - w(\mathcal{S}_\epsilon)$. The weight of each facility in $\mathcal{S}_\epsilon$ is set to zero. The facilities in $\mathcal{F}_\epsilon \setminus \mathcal{S}_\epsilon$ are removed from the instance. Let $\mathcal{I}' = (B', \mathcal{F} \setminus (\mathcal{F}_\epsilon \setminus \mathcal{S}_\epsilon), \mathcal{C}, c, w')$ be the modified instance. Since $\mathcal{S}$ is a feasible solution to $\mathcal{I}'$ it follows that $\mathrm{OPT}_{\mathcal{I}'} \leq \mathrm{OPT}_\mathcal{I}$. Therefore, the algorithm A from the statement outputs a solution $\mathcal{S}'$ whose cost is at most $\alpha \mathrm{OPT}_\mathcal{I}$. If $\mathcal{S}' \subseteq \mathcal{S}_\epsilon$ we are done since then $\mathcal{S}'$ is already feasible solution under the original weight w. Otherwise, let f_1, f_2 be the two heaviest facilities of $\mathcal{S}' \setminus \mathcal{S}_\epsilon$ where we set $f_2 = f_1$ if there is only one such facility. By the above-mentioned property of our algorithm, we have that $w'(\mathcal{S}' \setminus \{f_1, f_2\}) \leq B'$ and thus $w(\mathcal{S}' \setminus \{f_1, f_2\}) \leq B$. Since $f_1, f_2 \notin \mathcal{S}_\epsilon$ we have that $w(f_1)$ and $w(f_2)$ are bounded by ϵB. Hence the total weight of solution $\mathcal{S}'$ under the original weight function is $w(\mathcal{S}') \leq (1 + 2\epsilon)B$. $\qquad\square$

3.2 Computing and rounding a bi-point solution

Extending a similar result for the k-median [11], we can compute a so-called
bi-point solution, which is a convex combination of two integral pseudo-solutions.

Theorem 4 *We can compute in polynomial time two sets $\mathcal{F}_1$ and $\mathcal{F}_2$ of fa-
cilities and factors $a, b \geq 0$ such that $a + b = 1$, $w(\mathcal{F}_1) \leq B \leq w(\mathcal{F}_2)$,
$a \cdot w(\mathcal{F}_1) + b \cdot w(\mathcal{F}_2) \leq B$, and $a \cdot \text{cost}_{\mathcal{I}}(\mathcal{F}_1) + b \cdot \text{cost}_{\mathcal{I}}(\mathcal{F}_2) \leq 2 \cdot \text{OPT}_{\mathcal{I}}$.*

Proof We use the idea of Lagrangean relaxation to reduce the Knapsack me-
dian problem to the uncapacitated facility location problem. We would like
to get rid of problematic constraint $\sum_{i \in \mathcal{F}} w_i y_i \leq B$. Its violation will be pe-
nalized in the objective function by $\lambda(\sum_{i \in \mathcal{F}} w_i y_i - B)$, for some parameter
$\lambda \geq 0$. This penalty will favors solutions that obey the constraint. Our new
linear program is then

$$\min \quad \sum_{i \in \mathcal{F}, j \in \mathcal{C}} c_{ij} x_{ij} + \lambda \sum_{i \in \mathcal{F}} w_i y_i - \lambda B$$

$$\text{s.t.} \sum_{i \in \mathcal{F}} x_{ij} = 1 \qquad\qquad \forall j \in \mathcal{C}$$

$$x_{ij} \leq y_i \qquad\qquad \forall i \in \mathcal{F}, j \in \mathcal{C}$$

$$x_{i,j}, y_i \geq 0 \qquad\qquad \forall i \in \mathcal{F}, j \in \mathcal{C}$$

This LP gives a lower bound on OPT_f as each feasible solution to the
relaxation of the knapsack LP is also a feasible solution to the above LP
of no larger cost. In the above LP the term $-\lambda B$ in the objective function
is a constant. Therefore, this LP can be interpreted as a relaxation of the
uncapacitated facility location problem where each facility has i has an opening
cost λw_i. Note that increasing the parameter λ also increases the cost of the
facilities and will therefore generally lead to optimum solutions of smaller
facility weight (with respect to w). The idea of the algorithm is now to find
two values λ_1 and λ_2 for parameter λ, and two approximate solutions $\mathcal{F}_1$ and
$\mathcal{F}_2$ to the above facility location problem with these parameter settings such
that λ_1 and λ_2 are sufficiently close and such that $w(\mathcal{F}_1) \leq B \leq w(\mathcal{F}_2)$. It
can then be shown that a convex combination of these two solutions, called
bi-point solution, is a good approximation to the knapsack median problem.

Williamson and Shmoys (Section 7.7, pages 182–186 in [11]) prove an anal-
ogous theorem for the k-median problem, which arises when we set $w_i = 1$ for
all facilities i and $B = k$. We can extend this proof to the case of non-uniform
weights in a completely analogous manner. Moreover, instead of using the al-
gorithm of Jain and Vazirani [5] for facility location (which has approximation
ratio 3), we use a greedy algorithm of Jain et al. (Algorithm 2 in [4]) for facility
location achieving a factor of 2.   □

We will now give an algorithm which for a given KNAPSACK MEDIAN
instance $\mathcal{I} = (B, \mathcal{F}, \mathcal{C}, d, w)$ returns a pseudo-solution as in Lemma 10 with
cost $3.05 \text{OPT}_{\mathcal{I}}$.

We use Theorem 4 to obtain a bi-point solution of cost $2 \, \mathrm{OPT}_{\mathcal{I}}$. We will convert it into a pseudo-solution of cost 1.523 times bigger than the bi-point solution. Let $a\mathcal{F}_1 + b\mathcal{F}_2$ be the bi-point solution where $a + b = 1$, $w(\mathcal{F}_1) \leq B < w(\mathcal{F}_2)$ and $aw(\mathcal{F}_1) + bw(\mathcal{F}_2) = B$. For each client $j \in \mathcal{C}$ the closest elements in sets $\mathcal{F}_1$ and $\mathcal{F}_2$ are denoted by $i_1(j)$ and $i_2(j)$, respectively. Moreover, let $d_1(j) = c_{i_1(j)j}$ and $d_2(j) = c_{i_2(j)j}$. Then the (fractional) connection cost of j in our bi-point solution is $ad_1(j) + bd_2(j)$. In a similar way let $d_1 = \sum_{j \in \mathcal{C}} d_1(j)$ and $d_2 = \sum_{j \in \mathcal{C}} d_2(j)$. Then the bi-point solution has cost $ad_1 + bd_2$.

We consider two candidate solutions. In the first we just pick $\mathcal{F}_1$ which has cost bounded by $\frac{d_1}{ad_1 + bd_2} \leq \frac{1}{a + br_D}$, where $r_D = \frac{d_2}{d_1}$. This, multiplied by 2, gives our approximation factor.

To obtain the second candidate solution we use the concept of stars. For each facility $i \in \mathcal{F}_2$ define $\pi(i)$ to be the facility from set $\mathcal{F}_1$ which is closest to i. For a facility $i \in \mathcal{F}_1$ define star $\mathcal{S}_i$ with root i and leafs $L_i = \{i' \in \mathcal{F}_2 | \pi(i') = i\}$. Note that by the definition of stars, we have that any client j with $i_2(j) \in S_i$ has $c_{i_2(j)i} \leq c_{i_2(j)i_1(j)} = d_2(j) + d_1(j)$ and therefore $c_{ji} \leq c_{ji_2(j)} + c_{i_2(j)i} \leq 2d_2(j) + d_1(j)$.

The idea of the algorithm is to open for each star either its root or all of its leaves so that in total the budget is respected. We formulate this subproblem by means of an auxiliary LP. For any star $\mathcal{S}_i$ let $\delta(L_i) = \{j \in \mathcal{C} \mid i_2(j) \in S_i\}$. Consider a client $j \in \delta(L_i)$. If we open the root of $\mathcal{S}_i$ the connection cost of j is bounded by $2d_2(j) + d_1(j)$, but if we open the leaf $i_2(j) \in L_i$ we pay only $d_2(j)$ for connecting j. Thus, we save in total an amount of $\sum_{j \in \delta(L_i)} d_2(j) + d_1(j)$ when we open all leaves of $\mathcal{S}_i$ in comparison to opening just the root i. This leads us to the following linear programming relaxation where we introduce for each star $\mathcal{S}_i$ a variable x_i indicating whether we open the leaves of this star $(x_i = 1)$ or its root

$$\max \sum_{i \in \mathcal{F}_1} \sum_{j \in \delta(L_i)} (d_1(j) + d_2(j))x_i \quad \text{subject to} \tag{5}$$

$$\sum_{i \in \mathcal{F}_1} (w(S_i) - w_i)x_i \leq B - w(\mathcal{F}_1)$$

$$0 \leq x_i \leq 1 \quad \forall i \in \mathcal{F}_1 \, .$$

Now observe that this is a knapsack LP. Therefore, any optimum extreme point solution $\mathbf{x}$ to this LP has at most one fractional variable. Note that if we set $x_i = b$ for all $i \in \mathcal{F}_1$ we obtain a feasible solution to the above LP. Therefore the objective value of the above LP is lower bounded by $b(d_1 + d_2)$. We now open for all stars $\mathcal{S}_i$ with integral x_i either its root $(x_i = 0)$ or all of its leaves $(x_i = 1)$ according to the value of x_i. For the (single) star $\mathcal{S}_i$ where x_i is fractional we apply the following rounding procedure.

We always open i, the root of $\mathcal{S}_i$. To round the leaf set L_i, we set up another auxiliary knapsack LP similar to LP (5). In this LP, each leaf $i' \in L_i$ has a variable $\hat{x}_{i'}$ indicating if the facility is open $(\hat{x}_{i'} = 1)$ or not $(\hat{x}_{i'} = 0)$. For each leaf $i' \in L_i$ let $g_{i'} = \sum_{j \colon i_2(j) = i'} (d_1(j) + d_2(j))$ be its contribution to

LP (5). This gives rise to the following LP on the variables $\hat{x}_{i'}$ for all $i' \in L_i$. (The value x_i is a constant now.)

$$\max \sum_{i' \in L_i} g_{i'} \hat{x}_{i'} \quad \text{subject to}$$

$$\sum_{i' \in L_i} w_{i'} \hat{x}_{i'} \leq x_i \cdot w(L_i)$$

$$0 \leq \hat{x}_{i'} \leq 1 \quad \forall i' \in L_i \,.$$

Note that in the budget constraint of this LP we neglect the fact that the root is opened unconditionally, which causes a slight violation of the total budget bound B. Similar as for LP (5), we can compute an optimum extreme point solution $\hat{\mathbf{x}}$ which has at most one fractional variable $\hat{x}_{i'}$ and whose objective function value is lower bounded by $\sum_{j \in \delta(L_i)}(d_1(j) + d_2(j))x_i$. We open all $i' \in L_i$ with $\hat{x}_{i'} = 1$ and also the only fractional leaf. As a result, the overall set of opened facilities consists of a feasible solution and two additional facilities (namely the root i and the fractional leaf in L_i).

We will now analyze the cost of this solution. Both of the above knapsack LPs only reduce the connection cost in comparison to the original bipoint solution (or equivalently increase the saving with respect to the quantity $d_1 + 2d_2$), the total connection cost of the solution can be upper bounded by $d_1 + 2d_2 - b(d_1 + d_2) = (1 + a)d_2 + ad_1$.

The cost increase of the second algorithm with respect to the bi-point solution is at most

$$\frac{(1 + a)d_2 + ad_1}{(1 - a)d_2 + ad_1} = \frac{(1 + a)r_D + a}{(1 - a)r_D + a} ,$$

We always choose the better of the solutions of the two algorithms described above. Our approximation ratio is upper bounded by

$$\max_{\substack{r_D \geq 0 \\ a \in [0,1]}} \min \left\{ \frac{(1 + a)r_D + a}{(1 - a)r_D + a}, \frac{1}{a + r_D(1 - a)} \right\} \leq 1.523$$

This, multiplied by 2 gives our overall approximation ratio of 3.05.

Theorem 5 *For any $\epsilon > 0$, there is a bi-factor approximation algorithm for KM that computes a solution of weight $(1 + \epsilon)B$ and has a cost $3.05\mathrm{OPT}_\mathcal{I}$.*

Proof As argued above our algorithm computes a pseudo solution S of cost at most $3.05\mathrm{OPT}_\mathcal{I}$. Moreover, S consists of a feasible solution and two additional facilities. Hence, Lemma 10 implies the theorem. □

4 Discussion

The proof of Theorem 3 implies that for every (ϵ, δ)-sparse instance $\mathcal{I}$, there exists a solution $\mathcal{S}$ such that $\text{cost}_{\mathcal{I}}(\mathcal{S}) \leq (4+12/\delta)OPT_f + 3\epsilon OPT_{\mathcal{I}}$. Therefore, the integrality gap of $\mathcal{I}$ is at most $\frac{4+12/\delta}{1-3\epsilon}$. Unfortunately, our client-centric sparsification process inflates the approximation factor to at least $\frac{1+\delta}{1-\delta}$, so we must choose some $\delta < 1$ which balances this factor with that of Algorithm 3. In contrast, the facility-centric sparsification used in [8] incurs only a $1 + \epsilon$ factor in cost. We leave it as a open question whether the facility-centric version could also be used to get around the integrality gap of KM.

Our bi-factor approximation algorithm achieves a substantially smaller approximation ratio at the expense of slightly violating the budget by opening two extra facilities. We leave it as an open question, to obtain a pre- and post-processing in the flavor of Li and Svensson to turn this into an approximation algorithm. It seems even interesting to turn any bi-factor approximation into an approximation algorithm by losing only a constant factor in the approximation ratio. We also leave it as an open question to extend the third bi-point rounding procedure of Li and Svensson to knapsack median, which would give an improved result.

5 Acknowledgements

References

1. Byrka, J., Pensyl, T., Rybicki, B., Srinivasan, A., Trinh, K.: An improved approximation for k-median, and positive correlation in budgeted optimization. In: Proceedings of the Annual ACM-SIAM Symposium on Discrete Algorithms, (SODA), pp. 737–756 (2015)
2. Charikar, M., Guha, S., Tardos, É., Shmoys, D.B.: A constant-factor approximation algorithm for the k-median problem. In: ACM Symposium on Theory of Computing (STOC), pp. 1–10. ACM (1999)
3. Charikar, M., Li, S.: A dependent lp-rounding approach for the k-median problem. Automata, Languages, and Programming (ICALP) pp. 194–205 (2012)
4. Jain, K., Mahdian, M., Markakis, E., Saberi, A., Vazirani, V.V.: Greedy facility location algorithms analyzed using dual fitting with factor-revealing lp. J. ACM **50**(6), 795–824 (2003)
5. 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 **48**(2), 274–296 (2001). DOI 10.1145/375827.375845. URL http://doi.acm.org/10.1145/375827.375845
6. Krishnaswamy, R., Kumar, A., Nagarajan, V., Sabharwal, Y., Saha, B.: The matroid median problem. In: Proceedings of the annual ACM-SIAM Symposium on Discrete Algorithms (SODA), pp. 1117–1130. SIAM (2011)
7. Kumar, A.: Constant factor approximation algorithm for the knapsack median problem. In: Proceedings of the annual ACM-SIAM Symposium on Discrete Algorithms (SODA), pp. 824–832. SIAM (2012)

8. Li, S., Svensson, O.: Approximating k-median via pseudo-approximation. In: STOC, pp. 901–910 (2013)
9. Schrijver, A.: Combinatorial optimization: polyhedra and efficiency, vol. 24. Springer Science & Business Media (2003)
10. Swamy, C.: Improved approximation algorithms for matroid and knapsack median problems and applications. In: APPROX/RANDOM 2014, vol. 28, pp. 403–418 (2014)
11. Williamson, D.P., Shmoys, D.B.: The Design of Approximation Algorithms. Cambridge University Press (2011)