


Approximation Algorithms for Continuous Clustering and Facility Location Problems

Deeparnab Chakrabarty ✉🏠

Department of Computer Science, Dartmouth College, Hanover, NH, USA

Maryam Negahbani ✉🏠

Department of Computer Science, Dartmouth College, Hanover, NH, USA

Ankita Sarkar ✉🏠

Department of Computer Science, Dartmouth College, Hanover, NH, USA

Abstract

In this paper, we consider center-based clustering problems where C , the set of points to be clustered, lies in a metric space (X, d) , and the set X of candidate centers is potentially infinite-sized. We call such problems *continuous* clustering problems to differentiate them from the *discrete* clustering problems where the set of candidate centers is explicitly given. It is known that for many objectives, when one restricts the set of centers to C itself and applies an α_{dis} -approximation algorithm for the discrete version, one obtains a $\beta \cdot \alpha_{\text{dis}}$ -approximation algorithm for the continuous version via the triangle inequality property of the distance function. Here β depends on the objective, and for many objectives such as k -median, $\beta = 2$, while for some others such as k -means, $\beta = 4$. The motivating question in this paper is *whether this gap of factor β between continuous and discrete problems is inherent, or can one design better algorithms for continuous clustering than simply reducing to the discrete case as mentioned above?* In a recent SODA 2021 paper, Cohen-Addad, Karthik, and Lee prove a factor-2 and a factor-4 hardness, respectively, for the continuous versions of the k -median and k -means problems, even when the number of cluster centers is a constant. The discrete problem for a constant number of centers is easily solvable exactly using enumeration, and therefore, in certain regimes, the “ β -factor loss” seems unavoidable.

In this paper, we describe a technique based on the *round-or-cut* framework to approach continuous clustering problems. We show that, for the continuous versions of some clustering problems, we can design approximation algorithms attaining a better factor than the β -factor blow-up mentioned above. In particular, we do so for: the uncapacitated facility location problem with uniform facility opening costs (λ -UFL); the k -means problem; the individually fair k -median problem; and the k -center with outliers problem. Notably, for λ -UFL, where $\beta = 2$ and the discrete version is NP-hard to approximate within a factor of 1.27, we describe a 2.32-approximation for the continuous version, and indeed $2.32 < 2 \times 1.27$. Also, for k -means, where $\beta = 4$ and the best known approximation factor for the discrete version is 9, we obtain a 32-approximation for the continuous version, which is better than $4 \times 9 = 36$.

The main challenge one faces is that most algorithms for the discrete clustering problems, including the state of the art solutions, depend on Linear Program (LP) relaxations that become infinite-sized in the continuous version. To overcome this, we design new linear program relaxations for the continuous clustering problems which, although having exponentially many constraints, are amenable to the round-or-cut framework.

2012 ACM Subject Classification Theory of computation → Facility location and clustering

Keywords and phrases Approximation Algorithms, Clustering, Facility Location, Fairness, Outliers

Digital Object Identifier 10.4230/LIPIcs.ESA.2022.33

Related Version *Full Version:* <https://arxiv.org/abs/2206.15105> [14]

Funding The authors were supported by NSF award #2041920.

Acknowledgements We thank the reviewers for their comments, which helped us improve our exposition and exhibit a stronger connection to existing work.



© Deeparnab Chakrabarty, Maryam Negahbani, and Ankita Sarkar; licensed under Creative Commons License CC-BY 4.0

30th Annual European Symposium on Algorithms (ESA 2022).

Editors: Shiri Chechik, Gonzalo Navarro, Eva Rotenberg, and Grzegorz Herman; Article No. 33; pp. 33:1–33:15

Leibniz International Proceedings in Informatics



LIPICs Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

1 **Introduction**

Clustering is a ubiquitous problem arising in various areas ranging from data analysis to operations research. One popular class of clustering problems are the so-called *center-based clustering* problems where the quality of the clustering is determined by a function of the distances of every point in C to the “centers” of the clusters they reside in. Two extensively studied measures are the sum of these distances, with the resulting problem called the k -median problem, and the sum of squares of these distances, with the resulting problem called the k -means problem.

In most settings, these center-based clustering problems are NP-hard and one considers approximation algorithms for the same. Traditionally, however, approximation algorithms for these problems have been studied in *finite/discrete* metric spaces and, in fact, usually under the constraint that the set of centers, aka facilities, can be selected only from a prescribed subset $F \subseteq X$ of the metric space. Indeed, this model makes perfect sense when considering applications in operations research where the possible depot-locations may be constrained. These discrete problems have been extensively studied [33, 45, 16, 30, 17, 15, 35, 34, 5, 37, 39, 40, 8] over the last three decades. For instance, for the k -median problem, the best known approximation algorithm is a 2.675-approximation [9], while the best known hardness is $(1 + 2/e) \approx 1.74$ [34]. For the k -means problem, the best known approximation algorithm is a $(9 + \varepsilon)$ -approximation algorithm [1, 31, 37], while the best hardness is $1 + 8/e \approx 3.94$ [34].

Restricting to a finite metric space, however, makes the problem easier, and indeed many of the above algorithms in the papers mentioned above would be infeasible to implement if X were extremely large – for instance, if X were \mathbb{R}^m for some large dimension m , and the distance function were the ℓ_p -metric, for some p . On the other hand, it is reasonably easy to show using triangle inequality that if one considers opening centers from C itself, and thus reduces the problem to its discrete version, then one incurs a hit of a factor β in the approximation factor, where β is a constant depending on the objective function. In particular, if we look at the sum-of-distances objectives such as in k -median, then $\beta = 2$, while if one looks at the sum-of-squared-distances such as in k -means, then $\beta = 4$. Therefore, one immediately gets a 5.35-approximation for the *continuous k -median* problem and a $36 + \varepsilon$ -approximation for the *continuous k -means* problem. The question we investigate is

Is this factor β hit necessary between the continuous and discrete versions of center based clustering problems, or can one design better approximation algorithms for the continuous case?

It is crucial to note that when considering designing algorithms, we do not wish to make *any* assumptions on the underlying metric space (X, d) . For instance, we do not wish to assume $X = \mathbb{R}^m$ for some m . This is important, for we really want to compare ourselves with the β which is obtained using only the triangle-inequality and symmetry property of d . On the other hand, to exhibit that a certain algorithm does not work, any candidate metric space suffices.

Recently, in a thought-provoking paper [21], Cohen-Addad, Karthik, and Lee show that, unless $P = NP$, the k -median and k -means problem defined on $(\mathbb{R}^m, \ell_\infty)$ cannot have an approximation ratio better than 2 and 4, respectively, *even when k is a constant!* Since the discrete problems have trivial exact algorithms via enumeration when k is a constant, this seems to indicate that in certain cases the above factor β hit is unavoidable. Is it possible that the *inapproximability* of the continuous problem is indeed β times the inapproximability of the discrete version?

1.1 Our Results

Our main contribution is a direct approach towards the continuous versions of clustering problems. We apply this to the following clustering problems where we obtain a factor better than $\beta \cdot \alpha_{\text{dis}}$, where α_{dis} is the best known factor for the discrete version of the problem.

- In the continuous λ -UFL problem, a “soft” version of the continuous k -median problem, one is allowed to pick any number of centers but has to pay a parameter λ for each picked center. The objective is to minimize the sum of distances of points in C to picked centers plus the cost for opening these centers. Again, note that the centers can be opened anywhere in X . For the discrete version, where the only possible center locations are in C , there is a 1.488-approximation due to Li [39], and a hardness of approximation within a factor of 1.278 is known due to Guha and Khuller [30]. We describe a 2.32-approximation algorithm. Note that $2.32 < 2 \cdot 1.278$ and thus, for this problem, the inapproximability is *not* β times that of the discrete case. We also show how the reduction of [21] carries over, to prove a hardness of 2 for this problem.
- In the continuous k -means problem, we wish to minimize the sum of squares of distances of clients to the closest open center. Recall that for this problem we have $\beta = 4$, and thus one gets a 36-factor algorithm for the continuous k -means using the best known 9-factor [1, 31, 37] algorithm for the discrete problem. We describe an improved 32-approximation for the continuous k -means problem.
- For the continuous k -median problem, our techniques fall short of improving $2\times$ the best known approximation factor for the discrete k -median problem. On the other hand, we obtain better algorithms for the the *individually fair* or *priority* version of the continuous k -median problem. In this problem, every point $v \in C$ has a specified radius $r(v)$ and desires a center opened within this distance. The objective is the same as the k -median problem: minimize the sum of all the distances. This problem arises as a possible model [36, 13, 41, 47, 43, 7] in the study of fair clustering solutions, since the usual k -median algorithms may place certain clients inordinately far away. At a technical level, this problem is a meld of the k -median and the k -center problems; the latter is NP-hard, which forces one to look at *bicriterion* approximations. An (α, γ) -approximation would return a solution within α times the optimum but may connect v to a point as far as $\gamma r(v)$ away. Again, any (α, γ) -approximation for the *discrete* version where $X = C$ would imply a $(2\alpha, 2\gamma)$ -approximation for the continuous version.

The best discrete approximation is (8, 3) due to Vakilian and Yalçınır [47] which would imply an (16, 3)-approximation for the continuous version. We describe an (8, 8)-approximation for the continuous version of the problem.

- In the k -center with outliers (kCwO) problem, we are given a parameter $m \leq |C|$, and we need to serve only m of the clients. The objective is the maximum distance of a served client to its center. The k -center objective is one of the objectives for which most existing discrete algorithms can compare themselves directly with the continuous optimum. The 3-approximation algorithm in [17] for the kCwO problem is one such example. However, the best known algorithm for kCwO for the discrete case (when $X = C$) is a 2-approximation by Chakrabarty, Goyal, and Krishnaswamy [11] which proceeds via LP rounding, and does not give a 2-approximation for continuous kCwO. This was explicitly noted in a work by Ding, Yu, and Wang (“... unclear if the resulting approximation ratio for the problem in Euclidean space.”) [26], that describes a 2-approximation for kCwO in Euclidean space, however, violating the number of clients served. We give a proper 2-approximation for the continuous kCwO problem (with no assumptions on the metric space) with no violations.

1.2 Our Technical Insight

Most state of the art approximation algorithms for center-based clustering problems are based on LP relaxations where one typically has variables y_i for *every* potential location of a center. When the set X is large, this approach becomes infeasible. Our main technical insight, underlying all our results, is to use a different style of linear program with polynomially many variables but exponentially many constraints. We then use the *round-or-cut* framework to obtain our approximation factor. More precisely, given a *potential* solution to our program, we either “round” it to get a desired solution within the asserted approximation factor, or we find a *separating hyperplane* proving that this potential solution is infeasible. Once this hyperplane is fed to the ellipsoid algorithm [29], the latter generates another potential solution, and the process continues. Due to the ellipsoid method’s guarantees, we obtain our approximation factor in polynomial time.

For every client $v \in C$, our LP relaxation has variables of the form $y(v, r)$, indicating whether there is some point $x \in X$ in an r -radius around v which is “open” as a center. Throughout the paper we use r as a quantity varying “continuously”, but it can easily be discretized, with a loss of at most $\frac{1}{\text{poly}(n)}$, to arise from a set of size $\leq \text{poly}(n)$. Thus there are only polynomially many such variables. We add the natural “monotonicity” constraints: $y(v, r) \leq y(v, s)$ whenever $r \leq s$. Interestingly, for one of the applications, we also need the monotonicity constraints for non-concentric balls: if $B(v, r) \subseteq B(u, s)$, then we need $y(v, r) \leq y(u, s)$.

We have a variable C_v indicating the cost the client v pays towards the optimal solution. Next, we connect the C_v ’s and the $y(v, r)$ ’s in the following ways (when $\beta = 1$, and something similar when $\beta = 2$). One connection states that for any r , $C_v \geq r \cdot (1 - y(v, r))$ and we add these to our LP. For the last two applications listed above, this suffices. However, one can also state the stronger condition of $C_v \geq \int_0^\infty (1 - y(v, r)) dr$. Indeed, the weaker constraint is the “Markov-style inequality” version of the stronger constraint.

Our second set of constraints restrict the $y(v, r)$ ’s to be “not too large”. For instance, for the fair k -median or kCwO problems where we are only allowed k points from X , we assert that for *any* set \mathcal{B} of disjoint balls $B(v, r)$, we must have the sum of the respective $y(v, r)$ ’s to be at most k . This set of constraints is exponentially many, and this is the set of constraints that need the round-or-cut machinery. For the λ -UFL problem, we have that the sum of the $y(v, r)$ ’s scaled by λ plus the sum of the C_v ’s should be at most opt_g , which is a running guess of opt .

Once we set up the framework above, then we can *port many existing rounding* algorithms for the discrete clustering problems without much hassle. In particular, this is true for rounding algorithms which use the C_v ’s as the core driving force. For the continuous λ -UFL problem, we port the rounding algorithm from the paper [45] by Shmoys, Tardos, and Aardal. For the continuous k -means problem, we port the rounding algorithm from the paper [16] by Charikar, Guha, Shmoys, and Tardos. For the continuous fair k -median problem, we port the rounding algorithm from the paper [13] by Chakrabarty and Negahbani, which itself builds on the algorithm present in the paper [2] by Alamdari and Shmoys. For the continuous kCwO problem, we port the rounding algorithm present in the paper [11] by Chakrabarty, Goyal, and Krishnaswamy.

Our results fall short for the continuous k -median problem (without fairness), where we can port the rounding algorithm from the paper [16] and get a 6.67-approximation. This, however, does not improve upon the 5.35-factor mentioned earlier.

1.3 Other Related Works and Discussion

The continuous k -means and median problems have been investigated quite a bit in the specific setting when $X = \mathbb{R}^m$ and when $d(u, v)$ is the ℓ_2 distance. The paper [42] by Matoušek describes an $(1 + \varepsilon)$ -approximation (PTAS) that runs in time $O(n \log^k n \cdot \varepsilon^{-O(k^2 m)})$. This led to a flurry of results [32, 25, 38, 18, 27] on obtaining PTASes with better dependencies on k and m via the applications of coresets. There is a huge and growing literature on coresets, and we refer the interested reader to the paper [24] by Cohen-Addad, Saurpic, and Schwiegelshohn, and the references within, for more information. Another approach to the continuous k -means problem has been local search. The paper [37] which describes a $9 + \varepsilon$ -approximation was first stated for the geometric setting, however it also went via the discretization due to Matoušek [42] and suffered a running time of exponential dependency on the dimension. More recent papers [28, 22] described local-search based PTASes for metrics with doubling dimension D , with running time exponentially depending on D . These doubling metrics generalize (\mathbb{R}^m, ℓ_2) -metrics. However, none of the above ideas seem to suggest better constant factor approximations for the continuous k -median/means problem in the general case, and indeed even when $X = \mathbb{R}^m$ but m is part of the input.

The k -means problem in the metric space (\mathbb{R}^m, ℓ_2) , where m and k are not constants, has been studied extensively [46, 6, 19, 23, 1, 20], and is called the Euclidean k -means problem. The discrete version of this problem was proved APX-hard in 2000 [46], but the APX-hardness of the continuous version was proved much later, in 2015 [6]. More recently, the hardness results for both versions have been improved: the discrete Euclidean k -means problem is hard to approximate to factor 1.17, while the continuous problem is hard to approximate to factor 1.07 [19]. Moreover, under assumption of a complexity theoretic hypothesis called the Johnson coverage hypothesis, these numbers have been improved to 1.73 and 1.36, respectively [23]. On the algorithmic side, the discrete Euclidean k -means problem admits a better approximation ratio than the general case: a 6.36 approximation was described in 2017 [1], which was very recently improved to 5.912 [20].

We believe that our paper takes the first stab at getting approximation ratios better than $\beta \times$ the best discrete factor for the *continuous* clustering problems. Round-or-cut is a versatile framework for approximation algorithm design with many recent applications [44, 10, 3, 12, 4], and the results in our paper is yet another application of this paradigm. However, many questions remain. We believe that the most interesting question to tackle is the continuous k -median problem. The best known discrete k -median algorithms are, in fact, combinatorial in nature, and are obtained via applying the primal-dual/dual-fitting based methods [35, 34, 40, 9] on the discrete LP. However, their application still needs an explicit description of the facility set, and it is interesting to see if they can be *directly* ported to the continuous setting.

All the algorithms in our paper, actually *still* open centers from C . Even then, we are able to do better than simply reducing to the discrete case, because we do not commit to the β loss upfront, and instead round from a fractional solution that can open centers anywhere in X . This raises an interesting question for the k -median problem (or any other center based clustering problem): consider the potentially infinite-sized LP which has variables y_i for all $i \in X$, but restrict to the optimal solution which only is allowed to open centers from C . How big is this “integrality gap”? It is not too hard to show that for the k -median problem this is between 2 and 4. The upper bound gives hope we can get a true 4-approximation for the continuous k -median problem, but it seems one would need new ideas to obtain such a result.

Organization of Extended Abstract

Due to space constraints, in this extended abstract we have decided to focus on the continuous λ -UFL and the continuous fair k -median results since we believe that they showcase the technical ideas in this paper. Proofs of certain statements have also been deferred to the full version of the paper [14]. The description of the results on continuous k -means and continuous k -center with outliers can be found in the full version [14, Appendices B and C].

2 Preliminaries

Given a metric space (X, d) on points X with pairwise distances d , we use the notation $d(v, S) = \min_{i \in S} d(v, i)$ for $v \in X$ and $S \subseteq X$ to denote v 's distance to the set S .

► **Definition 1** (Continuous k -median (Cont- k -Med)). *The input is a metric space (X, d) , clients $C \subseteq X$, $|C| = n \in \mathbb{N}$, and $k \in \mathbb{N}$. The goal is to find $S \subseteq X$, $|S| = k$ minimizing $\text{cost}(S) := \sum_{v \in C} d(v, S)$.*

► **Definition 2** (Continuous Fair k -median (ContFair- k -Med)). *Given the Cont- k -Med input, plus fairness radii $r : C \rightarrow [0, \infty)$, the goal is to find $S \subseteq X$, $|S| = k$ such that $\forall v \in C$, $d(v, S) \leq r(v)$, minimizing $\text{cost}(S)$.*

In the Uncapacitated Facility Location (UFL) problem, the restriction of opening only k facilities is replaced by having a cost associated with opening each facility. When these costs are equal to the same value λ for all facilities, the problem is called λ -UFL.

► **Definition 3** (Continuous λ -UFL (Cont- λ -UFL)). *Given a metric space (X, d) , clients $C \subseteq X$, $|C| = n \in \mathbb{N}$, and $\lambda \geq 0$, find $S \subseteq X$ that minimizes the sum of “connection cost” $\text{cost}_C(S) := \sum_{v \in C} d(v, S)$ and “facility opening cost” $\text{cost}_F(S) := \lambda|S|$.*

Let $\Delta = \max_{u, v \in C} d(u, v)$ denote the diameter of a metric (X, d) . For $x \in X$, $0 \leq r \leq \Delta$, the ball of radius r around x is $B(x, r) := \{x' \in X \mid d(x', x) \leq r\}$. Throughout the paper, we use balls of the form $B(v, r)$ where v is a client and $r \in \mathbb{R}$. To circumvent the potentially infinite number of radii, the radii can be discretized into $I_\varepsilon = \{\varepsilon, 2\varepsilon, \dots, \lceil \Delta/\varepsilon \rceil \varepsilon\}$ for a small constant $\varepsilon = O(1/n^2)$. Thereupon, we can appeal to the following lemma to bound the size of I_ε by $O(n^5)$.

► **Lemma 4** (Rewording of Lemma 4.1, [1]). *Losing a factor of $(1 + \frac{100}{n^2})$, we can assume that for any $u, v \in C$, $1 \leq d(u, v) \leq n^3$.*

For simplicity of exposition, we present our techniques using radii in \mathbb{R} , and observe that discretizing to I_ε incurs an additive loss of at most $O(n\varepsilon) = O(1/n)$ in our guarantees. We also note that $\log \text{opt} \leq \log(n\Delta) = O(\log n)$ by the above, which enables us to efficiently binary-search over our guesses opt_g .

3 Continuous λ -UFL

We start this section with our 2.32-approximation for Cont- λ -UFL (Theorem 5). For this, we introduce a new linear programming formulation, and adapt the rounding algorithm of Shmoys-Tardos-Aardal to the new program. The resulting procedure exhibits our main ideas, and serves as a warm-up for the remaining sections. Also, in Section 3.2, we prove that it is NP-hard to approximate Cont- λ -UFL within a factor of $2 - o(1)$, using ideas due to Cohen-Addad, Karthik, and Lee [21]. This shows that the continuous version cannot be approximated as well as the discrete version, which has a best-known approximation factor of 1.463 [39].

3.1 Approximation algorithm

This subsection is dedicated to proving the following theorem:

► **Theorem 5.** *There is a polynomial time algorithm that, for an instance of Cont- λ -UFL with optimum opt , yields a solution with cost at most $(\frac{2}{1-e^{-2}} + \varepsilon)\text{opt} < 2.32 \cdot \text{opt}$. Here $\varepsilon = O(\frac{1}{n^2})$.*

We design the following linear program for Cont- λ -UFL. We use variables C_v for the connection cost of each client v , and $y(v, r)$ for the number of facilities opened within each ball of the form $B(v, r)$. We also use a guess of the optimum opt_g , which we will soon discuss how to obtain. Throughout, we use $y(B)$ as shorthand for $y(v, r)$ where $B = B(v, r)$.

$$\lambda \sum_{B \in \mathcal{B}} y(B) + \sum_{v \in C} C_v \leq \text{opt}_g \quad \forall \mathcal{B} \subseteq \{B(v, r)\}_{\substack{v \in C \\ r \in \mathbb{R}}} \text{ pairwise disjoint} \quad (\text{UFL})$$

$$\int_0^\infty (1 - y(v, r)) dr \leq C_v \quad \forall v \in C, r \in \mathbb{R} \quad (\text{UFL-1})$$

$$y(v, r) \leq y(v, r') \quad \forall v \in C, r, r' \in \mathbb{R} \text{ s.t. } r \leq r' \quad (\text{UFL-2})$$

$$y(v, r) \geq 0, C_v \geq 0 \quad \forall v \in C, r \in \mathbb{R}$$

Observe that, given a solution $S \subseteq X$ of cost at most opt_g , we can obtain a feasible solution of UFL as follows. For client $v \in C$, we set $C_v = d(v, S)$. For $v \in C, r \in \mathbb{R}$, we set $y(v, r) = 0$ for $r < d(v, S)$ and $y(v, r) = 1$ for $r \geq d(v, S)$.

Our approach is to round a solution (C, y) of UFL. Observe that there are polynomially many constraints of the form (UFL-1) and (UFL-2); hence, we can efficiently obtain a solution (C, y) that satisfies them. So for the remainder of this section, we assume that those constraints are satisfied. On the other hand, there are infinitely many constraints of type (UFL). This is why we employ a round-or-cut framework via the ellipsoid algorithm [29]. We begin with an arbitrary opt_g , and when ellipsoid asks us if a proposed solution (C, y) is feasible, we run the following algorithm.

The algorithm inputs $\alpha < 1$, and defines $r_\alpha(v)$ as the minimum radius at which client $v \in C$ has at least α mass of open facilities around it. First, all clients are deemed *uncovered* ($U = C$). Iteratively, the algorithm picks the j , i.e. the uncovered client, with the smallest $r_\alpha(j)$. j is put into the set $\text{Reps}_{(C, y), \alpha}$. Any client v within distance $r_\alpha(j) + r_\alpha(v)$ of j is considered a child of j and is now *covered*. When all clients are covered, i.e. $U = \emptyset$, the algorithm outputs $\text{Reps}_{(C, y), \alpha}$.

■ **Algorithm 1** Filtering for Cont- λ -UFL.

Input: A proposed solution $(\{C_v\}_{v \in C}, \{y(v, r)\}_{v \in C, r \in \mathbb{R}})$ for UFL, parameter $\alpha \in (e^{-2}, 1)$

- 1: $r_\alpha(v) \leftarrow \min\{r \in \mathbb{R} \mid y(v, r) \geq \alpha\}$ for all $v \in C$
- 2: $\text{Reps}_{(C, y), \alpha} \leftarrow \emptyset$ ▷ “representative” clients
- 3: $U \leftarrow C$ ▷ “uncovered” clients
- 4: **while** $U \neq \emptyset$ **do**
- 5: Pick $j \in U$ with minimum $r_\alpha(j)$
- 6: $\text{child}(j) \leftarrow \{v \in U \mid d(v, j) \leq r_\alpha(v) + r_\alpha(j)\}$
- 7: $U \leftarrow U \setminus \text{child}(j)$
- 8: $\text{Reps}_{(C, y), \alpha} \leftarrow \text{Reps}_{(C, y), \alpha} \cup j$
- 9: **end while**

Output: $\text{Reps}_{(C, y), \alpha}$

Notice that, by construction, the collection of balls $\{B(j, r_\alpha(j))\}_{j \in \text{Reps}_{(C,y),\alpha}}$ is pairwise disjoint. Hence, the following constraint, which we call $\text{Sep}_{(C,y),\alpha}$, is of the form (UFL):

$$\lambda \sum_{j \in \text{Reps}_{(C,y),\alpha}} y(j, r_\alpha(j)) + \sum_{v \in C} C_v \leq \text{opt}_g \quad (\text{Sep}_{(C,y),\alpha})$$

We will show that

► **Lemma 6.** *If (C, y) satisfies (UFL-1), (UFL-2), and $\text{Sep}_{(C,y),\alpha}$, then there exists a suitable $\alpha \in (e^{-2}, 1)$ for which the output of Algorithm 1 has cost at most $\frac{2}{1-e^{-2}} \text{opt}_g$.*

Thus, if we find that the desired approximation ratio is not attained, then it must be that $\text{Sep}_{(C,y),\alpha}$ was not satisfied, and we can pass it to ellipsoid as a separating hyperplane. If ellipsoid finds that the feasible region of our linear program is empty, then we increase opt_g and try again. Otherwise, we obtain a solution $\text{Reps}_{(C,y),\alpha}$ that attains the desired guarantees.

We now analyze Algorithm 1 to prove Lemma 6.

Proof of Lemma 6. For this proof, we will fix (C, y) , and refer to $\text{Reps}_{(C,y),\alpha}$ as Reps_α .

To prove a suitable α exists, assume α is picked uniformly at random from $(\beta, 1)$ for some $0 < \beta < 1$; we will see later that $\beta = e^{-2}$ is optimal. Take Reps_α , the output of Algorithm 1 on (C, y) . By definition of r_α , $\sum_{j \in \text{Reps}_\alpha} y(j, r_\alpha(j)) \geq \alpha |\text{Reps}_\alpha|$. Thus $\text{cost}_F(\text{Reps}_\alpha) = \lambda |\text{Reps}_\alpha| \leq \frac{1}{\alpha} \cdot \lambda \sum_{j \in \text{Reps}_\alpha} y(j, r_\alpha(j))$, which implies

$$\mathbf{Exp}[\text{cost}_F(\text{Reps}_\alpha)] \leq \frac{\ln(1/\beta)}{(1-\beta)} \lambda \sum_{j \in \text{Reps}_\alpha} y(j, r_\alpha(j)). \quad (1)$$

To bound the expected connection cost, take $v \in C$ and observe that, since all the clients are ultimately covered in Algorithm 1, there has to exist $j \in \text{Reps}_\alpha$ for which $v \in \text{child}(j)$. By construction of child , $d(v, j) \leq r_\alpha(v) + r_\alpha(j)$, which is at most $2r_\alpha(v)$ by our choice of j in Line 5. Thus, for any client v , we get $d(v, \text{Reps}_\alpha) \leq d(v, j) \leq 2r_\alpha(v)$. So we are left to bound $\mathbf{Exp}[r_\alpha(v)]$ for an arbitrary client $v \in C$.

We have that

$$\mathbf{Exp}[r_\alpha(v)] = \frac{1}{1-\beta} \int_\beta^1 r_\alpha(v) d\alpha \leq \frac{1}{1-\beta} \int_0^1 r_\alpha(v) d\alpha.$$

We notice that at $\alpha = y(v, r)$, $r_\alpha(v) = r$. Also, $r_0(v) = 0$. So given (UFL-2) for all balls $B(v, r)$ with $r \in \mathbb{R}$, we can apply a change of variable to the integral to get $\int_0^1 r_\alpha(v) d\alpha = \int_{r_0(v)}^{r_1(v)} (y(v, r_1(v)) - y(v, r)) dr \leq \int_0^\infty (1 - y(v, r)) dr \leq C_v$, where the last inequality is by (UFL-1). Thus we have $\mathbf{Exp}[r_\alpha(v)] \leq \frac{C_v}{1-\beta}$. Summing $d(v, \text{Reps}_\alpha)$ over all $v \in C$ we have

$$\mathbf{Exp}[\text{cost}_C(\text{Reps}_\alpha)] = \sum_{v \in C} \mathbf{Exp}[d(v, \text{Reps}_\alpha)] \leq 2 \sum_{v \in C} \mathbf{Exp}[r_\alpha(v)] \leq \frac{2}{1-\beta} \sum_{v \in C} C_v. \quad (2)$$

To balance cost_F from (1) and cost_C from (2), we set $\beta = e^{-2}$. The expected Cont- λ -UFL cost of Reps_α is, using $\text{Sep}_{(C,y),\alpha}$,

$$\mathbf{Exp}[\text{cost}_F(\text{Reps}_\alpha) + \text{cost}_C(\text{Reps}_\alpha)] \leq \frac{2}{1-e^{-2}} \left(\lambda \sum_{j \in \text{Reps}_\alpha} y(j, r_\alpha(j)) + \sum_{v \in C} C_v \right) \leq \frac{2 \cdot \text{opt}_g}{1-e^{-2}}.$$

Since the bound holds in expectation over a random α , there must exist an $\alpha \in (\beta, 1)$ that satisfies it deterministically. ◀

To obtain a suitable α , we can adapt the derandomization procedure from the discrete version [45]. The procedure relies on having polynomially many interesting radii; for this, we recall that while we have used $r \in \mathbb{R}$ for simplicity, our radii are actually $r \in I_\varepsilon$, $|I_\varepsilon| = O(n^5)$.

3.2 Hardness of approximation

Our hardness result for this problem is as follows:

► **Theorem 7.** *Given an instance of Cont- λ -UFL and $\varepsilon > 0$, it is NP-hard to distinguish between the following:*

- *There exists $S \subseteq X$ such that $\text{cost}_F(S) + \text{cost}_C(S) \leq (1 + 6\varepsilon)n$*
- *For any $S \subseteq X$, $\text{cost}_F(S) + \text{cost}_C(S) \geq (2 - \varepsilon)n$*

Thus we exhibit hardness of approximation up to a factor of $\frac{2-\varepsilon}{1+6\varepsilon}$, which tends to 2 as $\varepsilon \rightarrow 0$. Our reduction closely follows the hardness proof for Cont- k -Med [21]. The details appear in the full version of this paper [14, Appendix D].

4 Continuous Fair k -Median

The main result of this section is the following theorem.

► **Theorem 8.** *There exists a polynomial time algorithm for ContFair- k -Med that, for an instance with optimum cost opt , yields a solution with cost at most $8\text{opt} + \varepsilon$, in which, each client $v \in C$ is provided an open facility within distance $8r(v) + \varepsilon$ of itself. Here $\varepsilon = O\left(\frac{1}{n^2}\right)$.*

We create a round-or-cut framework, via the ellipsoid algorithm [29], that adapts the Chakrabarty-Negahbani algorithm [13] to the continuous setting. For this, we will modify the UFL linear program to suit ContFair- k -Med. As before, opt_g is a guessed optimum, C_v is the cost share of a client $v \in C$, and $y(v, r)$ represents the number of facilities opened in $B(v, r)$. There are two key modifications. First, we expand the monotonicity constraints of the form (UFL-2) to include *non-concentric* balls, which are crucial for adapting the fairness guarantee of Chakrabarty and Negahbani [13]. Second, we enforce the fairness constraints by requiring $y(v, r(v)) \geq 1$ for each client $v \in C$.

$$\sum_{v \in C} C_v \leq \text{opt}_g \quad (\text{LP})$$

$$\sum_{B \in \mathcal{B}} y(B) \leq k \quad \forall \mathcal{B} \subseteq \{B(v, r)\}_{\substack{v \in C \\ r \in \mathbb{R}}} \text{ pairwise disjoint} \quad (\text{LP-1})$$

$$\int_0^\infty (1 - y(v, r)) dr \leq C_v \quad \forall v \in C \quad (\text{LP-2})$$

$$y(u, r) \leq y(v, r') \quad \forall u, v \in C, r, r' \in \mathbb{R}, B(u, r) \subseteq B(v, r') \quad (\text{LP-3})$$

$$y(v, r(v)) \geq 1 \quad \forall v \in C \quad (\text{LP-4})$$

$$y(v, r) \geq 0, C_v \geq 0 \quad \forall v \in C, r \in \mathbb{R}$$

We will frequently use the following property of LP. See the full version of our paper [14, Appendix A.2] for the proof.

► **Lemma 9.** *Consider a solution (C, y) of LP. If for a client v , (C, y) satisfies all constraints of the form (LP-2) and (LP-3) involving v , then for any $r_0 \in \mathbb{R}$, $C_v \geq r_0(1 - y(v, r_0))$.*

As before, we will only worry about the constraints that are exponentially many. These are (LP-1). For this, we use ellipsoid [29]. Given a proposed solution (C, y) of LP, we construct $\text{Reps}_{(C,y)} \subseteq C$, as follows.

We first perform a filtering step. For each $v \in C$, we define $R(v) := \min r(v), 2C_v$. In the beginning, all clients are “uncovered” (i.e. $U = C$). In each iteration, let $j \in U$ be the uncovered client with the minimum $R(j)$; and add j to our set of “representatives” $\text{Reps}_{(C,y)}$. Any $v \in U$ within distance $2R(v)$ of j (including j itself) will be added to the set $\text{child}(j)$, and will be removed from U . After all clients are covered, i.e. $U = \emptyset$, the algorithm outputs $\text{Reps}_{(C,y)}$. For a formal description of this algorithm, see the full version [14, Appendix A.1].

For a $j \in \text{Reps}_{(C,y)}$, let $s(j)$ be the closest client to j in $\text{Reps}_{(C,y)} \setminus \{j\}$. Let $a(j) := \frac{d(j,s(j))}{2}$. So the collection of balls $\{B(j, a(j))\}_{j \in \text{Reps}_{(C,y)}}$ is pairwise disjoint, and the following constraint, which we call $\text{Sep}_{(C,y)}$, is of the form (LP-1).

$$\sum_{j \in \text{Reps}_{(C,y)}} y(j, a(j)) \leq k \quad (\text{Sep}_{(C,y)})$$

We have that

► **Lemma 10.** *If (C, y) satisfies (LP-2)-(LP-4) and $\text{Sep}_{(C,y)}$, then*

1. $\forall j \in \text{Reps}_{(C,y)}, y(j, a(j)) \geq \frac{1}{2}$
2. $|\text{Reps}_{(C,y)}| \leq 2k$

Proof. Fix $j \in \text{Reps}_{(C,y)}$. By construction of $\text{Reps}_{(C,y)}$,

$$a(j) = \frac{d(j, s(j))}{2} \geq R(j) \quad (3)$$

So if $R(j) = r(j)$, then by (LP-3) and (LP-4), $y(j, a(j)) \geq y(j, r(j)) \geq 1$. Else $R(j) = 2C_j$. By Lemma 9, $C_j \geq a(j) (1 - y(j, a(j)))$.

If $C_j = 0$, then this implies $y(j, a(j)) \geq 1$. Otherwise, substituting $a(j)$ by $R(j)$ from (3) and setting $R(j) = 2C_j$ gives $C_j \geq 2C_j (1 - y(j, a(j)))$, i.e. $y(j, a(j)) \geq \frac{1}{2}$. Now, by $\text{Sep}_{(C,y)}$, we have $k \geq \sum_{j \in \text{Reps}_{(C,y)}} y(j, a(j)) \geq \frac{1}{2} \cdot |\text{Reps}_{(C,y)}|$. ◀

So if we find that $|\text{Reps}_{(C,y)}| > 2k$, then $\text{Sep}_{(C,y)}$ must be violated, and we can pass it to ellipsoid as a separating hyperplane. Hence in polynomial time, we either find that our feasible region is empty, or we get (C, y) and $\text{Reps}_{(C,y)}$ such that (C, y) satisfies (LP), (LP-2)-(LP-4), and $\text{Sep}_{(C,y)}$. In the first case, we increase opt_g and try again. In the latter case, we round (C, y) further to attain our desired approximation ratios, via a rounding algorithm that we will now describe. This algorithm focuses on $\text{Reps}_{(C,y)}$ and ignores other clients, as justified by the following lemma.

► **Lemma 11.** *$S \subseteq X$ be a solution to ContFair- k -Med. Consider a proposed solution (C, y) of LP that satisfies (LP). Then $\sum_{v \in C} d(v, S) \leq \sum_{j \in \text{Reps}_{(C,y)}} |\text{child}(j)| d(j, S) + 4\text{opt}_g$.*

The proof closely follows from a standard technique for the discrete version [16, 13]. We provide the proof in the full version [14, Appendix A.3].

Our algorithm will also ignore facilities outside $\text{Reps}_{(C,y)}$, so our solution will be a subset of $\text{Reps}_{(C,y)}$. For the remainder of this section, we fix (C, y) , and refer to $\text{Reps}_{(C,y)}$ as Reps . We write the following polynomial-sized linear program, DLP, where Reps are the only clients and the only facilities. The objective function of DLP is a lower bound on $\sum_{j \in \text{Reps}} |\text{child}(j)| d(j, S)$, so hereafter we compare our output with DLP. We do not include fairness constraints in this program, and we will see later that it is not necessary to do so.

In DLP, the variables z_i for each $i \in \text{Reps}$ denote whether i is open as a facility. The variables x_{ij} for $i, j \in \text{Reps}$ denote whether the client j uses the facility i .

$$\text{minimize } \sum_{j \in \text{Reps}} |\text{child}(j)| \sum_{i \in \text{Reps}} x_{ij} d(j, i) \quad (\text{DLP})$$

$$\sum_{i \in \text{Reps}} z_i \leq k \quad (\text{DLP-1})$$

$$\sum_{i \in \text{Reps}} x_{ij} = 1 \quad \forall j \in \text{Reps} \quad (\text{DLP-2})$$

$$x_{ij} \leq y_i \quad \forall i, j \in \text{Reps} \quad (\text{DLP-3})$$

$$x_{ij} \geq 0, z_i \geq 0 \quad \forall i, j \in \text{Reps}$$

We will now round (C, y) to an integral solution of DLP. Our first step is to convert (C, y) to a fractional solution (\bar{x}, \bar{z}) of DLP. To do this, for each $j \in \text{Reps}$, we consolidate the y -mass in $B(j, a(j))$ onto j , i.e. we set $\bar{z}_j = y(j, a(j))$. By Lemma 10.1, each \bar{z}_j is then at least $\frac{1}{2}$. This allows j to use only itself and $s(j)$ as its fractional facilities.

■ **Algorithm 2** Consolidation for ContFair- k -Med.

Input: A proposed solution $(\{C_v\}_{v \in C}, \{y(v, r)\}_{v \in C, r \in I_\varepsilon})$ for LP, and Reps as defined above

- 1: **for** $j \in \text{Reps}$ **do**
- 2: $s(j) \leftarrow \arg \min_{v \in \text{Reps} \setminus j} d(j, v)$
- 3: $a(j) \leftarrow d(j, s(j))/2$
- 4: $\bar{z}_j \leftarrow \min \{y(j, a(j)), 1\}$
- 5: $\bar{x}_{jj} \leftarrow \bar{z}_j$
- 6: $\bar{x}_{js(j)} \leftarrow 1 - \bar{z}_j$
- 7: **end for**

Output: (\bar{x}, \bar{z})

► **Lemma 12.** $\forall j \in \text{Reps}, \bar{z}_j \geq \frac{1}{2}$, and (\bar{x}, \bar{z}) is a feasible solution of DLP with cost at most 2opt_g .

Proof. For a $j \in \text{Reps}$, if $y(j, a(j)) = 1$ then $\bar{z}_j = 1$. Otherwise, by Lemma 10.1, $\bar{z}_j = y(j, a(j)) \geq \frac{1}{2}$.

Hence $1 - \bar{z}_j \leq \frac{1}{2} \leq \bar{z}_{s(j)}$, which implies feasibility by construction and $\text{Sep}_{(C, y)}$. It also implies that $\sum_{i \in \text{Reps}} \bar{x}_{ij} d(j, i) = (1 - \bar{z}_j) d(j, s(j))$. If $y(j, a(j)) = 1$, then the RHS above is $0 \leq 2C_j$. Otherwise $\sum_{i \in \text{Reps}} \bar{x}_{ij} d(j, i) = (1 - y(j, a(j))) d(j, s(j)) = 2(1 - y(j, a(j))) a(j) \leq 2C_j$ where the last inequality follows from Lemma 9. Multiplying by $|\text{child}(j)|$ and summing over all $j \in \text{Reps}$, we have by construction of Reps ,

$$\sum_{j \in \text{Reps}} |\text{child}(j)| \sum_{i \in \text{Reps}} \bar{x}_{ij} d(j, i) \leq 2 \sum_{j \in \text{Reps}} |\text{child}(j)| C_j \leq 2 \sum_{j \in \text{Reps}} \sum_{v \in \text{child}(j)} C_v = 2 \sum_{v \in C} C_v$$

which is at most 2opt_g by (LP). ◀

Now, to round (\bar{x}, \bar{z}) to an integral solution, we appeal to an existing technique [16, 13]. We state the relevant result here, and provide the proof in the full version [14, Appendix A.4].

► **Lemma 13** ([16, 13]). *Let (\bar{x}, \bar{z}) be a feasible solution of DLP with cost at most 2opt_g , such that $\forall j \in \text{Reps}, \bar{z}_j \geq \frac{1}{2}$. Then there exists a polynomial time algorithm that produces $S \subseteq \text{Reps}$ such that*

1. $|S| = k$;
2. If $\bar{z}_j = 1$, then $j \in S$;
3. $\forall j \in \text{Reps}$, at least one of $j, s(j)$ is in S ; and
4. $\sum_{j \in \text{Reps}} |\text{child}(j)| d(j, S) \leq 4\text{opt}_g$.

Thus, by Lemma 11, we have shown that $\sum_{v \in C} d(v, S) \leq \sum_{j \in \text{Reps}} |\text{child}(j)| \sum_{i \in S} d(j, S) + 4\text{opt}_g \leq 8\text{opt}_g$. Now we show the fairness ratio, adapting a related result [13, Lemma 3] from the discrete version. This is where we crucially require the monotonicity constraints (LP-3) for *non-concentric* balls.

► **Lemma 14.** $\forall v \in C, d(v, S) \leq 8r(v)$.

Proof. Fix $v \in C$, and let $j \in \text{Reps}$ such that $v \in \text{child}(j)$. By construction of $\text{child}(j)$,

$$d(v, j) \leq 2R(v) \leq 2r(v) \tag{4}$$

So if $j \in S$, then we are done. Otherwise, by Lemma 13.2, $\bar{z}_j < 1$, i.e. by Algorithm 2, $y(j, a(j)) < 1$. But by the fairness constraints (LP-4), $y(v, r(v)) \geq 1$. So by the monotonicity constraints (LP-3), $B(v, r(v)) \not\subseteq B(j, a(j))$, as otherwise we would have $1 \leq y(v, r(v)) \leq y(j, a(j)) < 1$, a contradiction.

So fix $w \in B(v, r(v)) \setminus B(j, a(j))$. We have

$$a(j) < d(j, w), \quad d(v, w) \leq r(v) \tag{5}$$

By Lemma 13.3, either $j \in S$ or $s(j) \in S$, so

$$\begin{aligned} d(j, S) &\leq d(j, s(j)) = 2a(j) < 2d(j, w) && \dots \text{ by (5)} \\ &\leq 2(d(v, j) + d(v, w)) \leq 2(2r(v) + r(v)) && \dots \text{ by (4) and (5)} \\ &= 6r(v) \end{aligned}$$

So, by (4), $d(v, S) \leq d(v, j) + d(j, S) \leq 2r(v) + 6r(v) = 8r(v)$. ◀

Thus we have proved Theorem 8.

We observe here that, by the simple reduction of setting all $r(v)$'s to ∞ , Theorem 8 implies a solution of cost $8\text{opt} + \varepsilon$ for *Cont- k -Med*. We improve this ratio via an improved rounding procedure by Charikar, Guha, Shmoys, and Tardos [16], which rounds (\bar{x}, \bar{z}) such that $\sum_{j \in \text{Reps}} |\text{child}(j)| d(j, S) \leq \frac{8}{3}\text{opt}_g$, instead of the 4opt_g that we obtain above. This yields:

► **Corollary 15.** *There exists a polynomial time algorithm for Cont- k -Med that, on an instance with optimum cost opt , yields a solution of cost at most $6\frac{2}{3}\text{opt} + \varepsilon$.*

This improved rounding, however, no longer guarantees to open either j or $s(j)$ for each $j \in \text{Reps}$. Such a guarantee (Lemma 13.3) is crucial to our fairness bound in Lemma 14. So the improvement is not naively adaptable to *ContFair- k -Med*.

References

- 1 Sara Ahmadian, Ashkan Norouzi-Fard, Ola Svensson, and Justin Ward. Better guarantees for k -means and euclidean k -median by primal-dual algorithms. In *2017 IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 61–72, 2017.
- 2 Soroush Alamdari and David B. Shmoys. A bicriteria approximation algorithm for the k -center and k -median problems. In *Proc., Workshop on Approximation and Online Algorithms (WAOA)*, pages 66–75. Springer, 2017.

- 3 Hyung-Chan An, Mohit Singh, and Ola Svensson. LP-based algorithms for capacitated facility location. *SIAM Journal on Computing (SICOMP)*, 46(1):272–306, 2017. Preliminary version in FOCS 2014.
- 4 Georg Anegg, Haris Angelidakis, Adam Kurpisz, and Rico Zenklusen. A technique for obtaining true approximations for k-center with covering constraints. *Math. Program.*, 192(1):3–27, 2022. doi:10.1007/s10107-021-01645-y.
- 5 V. Arya, N. Garg, R. Khandekar, A. Meyerson, K. Munagala, and V. Pandit. Local Search Heuristics for k-Median and Facility Location Problems. *SIAM Journal on Computing (SICOMP)*, 33(3):544–562, 2004.
- 6 Pranjal Awasthi, Moses Charikar, Ravishankar Krishnaswamy, and Ali Kemal Sinop. The hardness of approximation of euclidean k-means. *CoRR*, 2015. arXiv:1502.03316.
- 7 Tanvi Bajpai, Deeparnab Chakrabarty, Chandra Chekuri, and Maryam Negahbani. Revisiting Priority k-Center: Fairness and Outliers. In *Proc., International Colloquium on Automata, Languages and Programming (ICALP)*, pages 21:1–21:20, 2021.
- 8 Jaroslav Byrka, Fabrizio Grandoni, Thomas Rothvoss, and Laura Sanità. Steiner tree approximation via iterative randomized rounding. *Journal of the ACM*, 60(1):1–33, 2013. doi:10.1145/2432622.2432628.
- 9 Jarosław Byrka, Thomas Pensyl, Bartosz Rybicki, Aravind Srinivasan, and Khoa Trinh. An improved approximation for k-median and positive correlation in budgeted optimization. *ACM Trans. Algorithms*, 2017.
- 10 Deeparnab Chakrabarty, Chandra Chekuri, Sanjeev Khanna, and Nitish Korula. Approximability of capacitated network design. *Algorithmica*, 72(2):493–514, 2015.
- 11 Deeparnab Chakrabarty, Prachi Goyal, and Ravishankar Krishnaswamy. The non-uniform k-center problem. *ACM Trans. Algorithms*, 2020. Preliminary version in ICALP 2016.
- 12 Deeparnab Chakrabarty and Maryam Negahbani. Generalized center problems with outliers. *ACM Trans. Algorithms*, 2019.
- 13 Deeparnab Chakrabarty and Maryam Negahbani. Better algorithms for individually fair k-clustering. *Adv. in Neu. Inf. Proc. Sys. (NeurIPS)*, 2021.
- 14 Deeparnab Chakrabarty, Maryam Negahbani, and Ankita Sarkar. Approximation algorithms for continuous clustering and facility location problems. *CoRR*, 2022. arXiv:2206.15105.
- 15 Moses Charikar and Sudipto Guha. Improved Combinatorial Algorithms for the Facility Location and k-Median Problems. In *Proc., IEEE Symposium on Foundations of Computer Science (FOCS)*, 1999.
- 16 Moses Charikar, Sudipto Guha, Éva Tardos, and David B. Shmoys. A constant-factor approximation algorithm for the k-median problem. *Journal of Computer and System Sciences*, 65(1):129–149, 2002.
- 17 Moses Charikar, Samir Khuller, David M. Mount, and Giri Narasimhan. Algorithms for facility location problems with outliers. In *Proc., ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 642–651, 2001.
- 18 Ke Chen. On k-median clustering in high dimensions. In *Proc., ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1177–1185, 2006.
- 19 Vincent Cohen-Addad and Karthik C.S. Inapproximability of clustering in lp metrics. In *2019 IEEE 60th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 519–539, 2019. doi:10.1109/FOCS.2019.00040.
- 20 Vincent Cohen-Addad, Hossein Esfandiari, Vahab Mirrokni, and Shyam Narayanan. Improved approximations for Euclidean k-means and k-median, via nested quasi-independent sets. In *Proceedings of the 54th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2022*, pages 1621–1628, New York, NY, USA, 2022. Association for Computing Machinery. doi:10.1145/3519935.3520011.
- 21 Vincent Cohen-Addad, C. S. Karthik, and Euiwoong Lee. On approximability of clustering problems without candidate centers. In *Proc., ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 2635–2648, 2021.

- 22 Vincent Cohen-Addad, Philip N Klein, and Claire Mathieu. Local search yields approximation schemes for k -means and k -median in euclidean and minor-free metrics. *SIAM Journal on Computing (SICOMP)*, 48(2):644–667, 2019.
- 23 Vincent Cohen-Addad, Karthik C. S., and Euiwoong Lee. Johnson coverage hypothesis: Inapproximability of k -means and k -median in ℓ_p -metrics. In *Proceedings of the 2022 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1493–1530, 2022. doi:10.1137/1.9781611977073.63.
- 24 Vincent Cohen-Addad, David Saulpic, and Chris Schwiegelshohn. A new coresets framework for clustering. In *Proc., ACM Symposium on Theory of Computing (STOC)*, pages 169–182, 2021.
- 25 W Fernandez De La Vega, Marek Karpinski, Claire Kenyon, and Yuval Rabani. Approximation schemes for clustering problems. In *Proc., ACM Symposium on Theory of Computing (STOC)*, pages 50–58, 2003.
- 26 Hu Ding, Haikuo Yu, and Zixiu Wang. Greedy strategy works for k -center clustering with outliers and coresets construction. In *Proc., European Symposium on Algorithms*, pages 40:1–40:16, 2019.
- 27 Dan Feldman, Morteza Monemizadeh, and Christian Sohler. A PTAS for k -means clustering based on weak coresets. In *Proceedings of the 5th Annual Symposium on Computational Geometry (SoCG)*, pages 11–18, 2007.
- 28 Zachary Friggstad, Mohsen Rezapour, and Mohammad R Salavatipour. Local search yields a PTAS for k -means in doubling metrics. *SIAM Journal on Computing (SICOMP)*, 48(2):452–480, 2019.
- 29 Martin Grötschel, László Lovász, and Alexander Schriver. *Geometric Algorithms and Combinatorial Optimization*. Springer, Berlin, Heidelberg, 1993.
- 30 Sudipto Guha and Samir Khuller. Greedy strikes back: Improved facility location algorithms. *Journal of Algorithms*, 31(1):228–248, 1999.
- 31 Anupam Gupta and Kanat Tangwongsan. Simpler analyses of local search algorithms for facility location. *CoRR*, 2008. arXiv:0809.2554.
- 32 S. Har-Peled and S. Mazumdar. Coresets for k -means and k -median clustering and their applications. In *Proc., ACM Symposium on Theory of Computing (STOC)*, pages 291–300, 2004.
- 33 Dorit S. Hochbaum and David B. Shmoys. A best possible heuristic for the k -center problem. *Math. Oper. Res.*, 10(2):180–184, 1985. doi:10.1287/moor.10.2.180.
- 34 Kamal Jain, Mohammad Mahdian, Evangelos Markakis, Amin Saberi, and Vijay V Vazirani. Greedy facility location algorithms analyzed using dual fitting with factor-revealing LP. *Journal of the ACM*, 50(6):795–824, 2003.
- 35 Kamal Jain and Vijay V. Vazirani. Approximation algorithms for metric facility location and k -median problems using the primal-dual schema and Lagrangean relaxation. *Journal of the ACM*, 48(2):274–296, 2001.
- 36 Christopher Jung, Sampath Kannan, and Neil Lutz. Service in your neighborhood: Fairness in center location. In *Proceedings, Foundations of Responsible Computing, FORC 2020*, volume 156, pages 5:1–5:15, 2020.
- 37 Tapas Kanungo, David M Mount, Nathan S Netanyahu, Christine D Piatko, Ruth Silverman, and Angela Y Wu. A local search approximation algorithm for k -means clustering. *Computational Geometry*, 28(2-3):89–112, 2004.
- 38 A. Kumar, Y. Sabharwal, and S. Sen. A simple linear time $(1 + \epsilon)$ -approximation algorithm for k -means clustering in any dimensions. In *Proc., IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 454–462, 2004.
- 39 Shi Li. A 1.488 approximation algorithm for the uncapacitated facility location problem. *Information and Computation*, 222:45–58, 2013.
- 40 Shi Li and Ola Svensson. Approximating k -median via pseudo-approximation. *SIAM Journal on Computing (SICOMP)*, 45(2):530–547, 2016.

- 41 Sepideh Mahabadi and Ali Vakilian. (individual) fairness for k -clustering. In *Proc., International Conference on Machine Learning (ICML)*, pages 7925–7935, 2020.
- 42 Jiri Matoušek. On approximate geometric k -clustering. *Discrete & Computational Geometry*, 24(1):61–84, 2000.
- 43 Ján Plesník. A heuristic for the p -center problems in graphs. *Discrete Applied Mathematics*, 17(3):263–268, 1987.
- 44 Robert D. Carr, Toshihiro Fujito, Goran Konjevod, and Ojas Parekh. A $2\frac{1}{10}$ -approximation Algorithm for a Generalization of the Weighted Edge-Dominating Set Problem. *J. Comb. Optim.*, 2001. Preliminary version appeared in *Proc. 8th ESA*, pages 132–142, 2000.
- 45 David B. Shmoys, Éva Tardos, and Karen Aardal. Approximation algorithms for facility location problems (extended abstract). In *Proc., ACM Symposium on Theory of Computing (STOC)*, 1997.
- 46 Luca Trevisan. When hamming meets euclid: The approximability of geometric tsp and steiner tree. *SIAM Journal on Computing*, 30(2):475–485, 2000. doi:10.1137/S0097539799352735.
- 47 Ali Vakilian and Mustafa Yalciner. Improved approximation algorithms for individually fair clustering. In Gustau Camps-Valls, Francisco J. R. Ruiz, and Isabel Valera, editors, *Proceedings of The 25th International Conference on Artificial Intelligence and Statistics*, volume 151 of *Proceedings of Machine Learning Research*, pages 8758–8779. PMLR, 28–30 March 2022. URL: <https://proceedings.mlr.press/v151/vakilian22a.html>.