Approximation Algorithms for Socially Fair Clustering

Yury Makarychev   YURY@TTIC.EDU and **Ali Vakilian**   VAKILIAN@TTIC.EDU

Toyota Technological Institute at Chicago, 6045 S Kenwood Ave, Chicago, IL 60637, USA

Abstract

We present an $(e^{O(p)}\frac{\log \ell}{\log \log \ell})$-approximation algorithm for socially fair clustering with the ℓ_p-objective. In this problem, we are given a set of points in a metric space. Each point belongs to one (or several) of ℓ groups. The goal is to find a k-medians, k-means, or, more generally, ℓ_p-clustering that is simultaneously good for all of the groups. More precisely, we need to find a set of k centers C so as to minimize the maximum over all groups j of $\sum_{u \text{ in group } j} d(u, C)^p$.

The socially fair clustering problem was independently proposed by Abbasi, Bhaskara, and Venkatasubramanian (2021) and Ghadiri, Samadi, and Vempala (2021). Our algorithm improves and generalizes their $O(\ell)$-approximation algorithms for the problem.

The natural LP relaxation for the problem has an integrality gap of $\Omega(\ell)$. In order to obtain our result, we introduce a strengthened LP relaxation and show that it has an integrality gap of $\Theta(\frac{\log \ell}{\log \log \ell})$ for a fixed p. Additionally, we present a bicriteria approximation algorithm, which generalizes the bicriteria approximation of Abbasi et al. (2021).

1. Introduction

Due to increasing use of machine learning in decision making, there has been an extensive line of research on the societal aspects of algorithms (Galindo and Tamayo, 2000; Kleinberg et al., 2017; Chouldechova, 2017; Dressel and Farid, 2018). The goal of research of this area is to understand, on one hand, *what it means for an algorithm to be fair and how to formally define the fairness requirements* and, on the other hand, *how to design efficient algorithms that meet the fairness requirements*. By now, there have been a rich amount of studies on both fronts for different machine learning tasks (Dwork et al., 2012; Feldman et al., 2015; Hardt et al., 2016; Chierichetti et al., 2017; Kleindessner et al., 2019; Jung et al., 2020; Har-Peled and Mahabadi, 2019; Aumüller et al., 2020). We refer the reader to (Chouldechova and Roth, 2018; Kearns and Roth, 2019) for an overview of different notions of fairness and their computational aspects.

In this work, we study clustering under the notion of group fairness called *social fairness* or *equitable group representation*, introduced independently by Abbasi, Bhaskara, and Venkatasubramanian (2021) and Ghadiri, Samadi, and Vempala (2021). In this framework, we need to cluster a dataset with points coming from a number of groups. Ghadiri et al. observed that both standard clustering algorithms and existing fair clustering algorithms (e.g., those that ensure that various groups are fairly represented in all clusters; see (Chierichetti et al., 2017)) incur higher clustering costs for certain protected groups (e.g., groups that are defined by a sensitive attribute such as gender and race). This lack of fairness has motivated the study of clustering that minimizes the maximum clustering cost across different demographic groups. The objective of socially fair clustering was previously studied by Anthony, Goyal, Gupta, and Nagarajan (2010) in the context of *robust clustering*. In this setting, a set of possible scenarios is provided and the goal is to find a solution that is simultaneously good for all scenarios.

Definition 1 (Socially Fair ℓ_p-Clustering) *We are given a metric space (P, d) on n points, ℓ groups of points $P_1, \cdots, P_\ell \subseteq P$, and the desired number of centers k. Additionally, we are given non-negative demands or weights $w_j : P_j \to \mathbb{R}$ for points in each group P_j. We define the ℓ_p-cost of group P_j w.r.t. a set of centers $C \subseteq P$ as*

$$\text{cost}(C, w_j) = \sum_{p \in P_j} w_j(p) \cdot d(p, C)^p,$$

where $d(p, C) = \min_{c \in C} d(p, c)$. In socially fair ℓ_p-clustering, the goal is to pick a set of k centers $C \subseteq P$ so as to minimize $\max_{j \in [\ell]} \text{cost}(C, w_j)$, which we call the fair cost:

$$\text{fair-cost}(C) = \text{fair-cost}(C, \{w_j\}_{j \in [\ell]}) := \max_{j \in [\ell]} \sum_{p \in P_j} w_j(p) \cdot d(p, C)^p. \tag{1}$$

The problem was first studied in the context of robust clustering. Anthony, Goyal, Gupta, and Nagarajan (2010) introduced it under the name "robust k-medians" (for $p = 1$) and designed an $O(\log |P| + \log \ell)$-approximation algorithm for it (for $p = 1$). Recently, Abbasi, Bhaskara, and Venkatasubramanian (2021)[1] and Ghadiri, Samadi, and Vempala (2021) considered this problem in the context of social fairness. They studied this problem for the most interesting cases, when $p \in \{1, 2\}$ and $p = 2$, respectively; the papers assumed that $w_j = 1/|P_j|$. In these special cases, the problem is called socially fair k-medians ($p = 1$) and socially fair k-means ($p = 2$). Both papers presented $O(\ell)$-approximation algorithms for the variants of fair clustering they study. Further, Abbasi et al. gave a bicriteria $(2/\gamma, 1/(1 - \gamma))$-approximation algorithm for fair k-medians and k-means. Their algorithm finds a solution with at most $k/(1 - \gamma)$ centers, whose cost is at most $2/\gamma$ times the optimal cost for k centers. Also, Ghadiri et al. designed a fair variant of Lloyd's heuristic for fair k-means clustering. Further, Bhattacharya, Chalermsook, Mehlhorn, and Neumann (2014) showed that it is hard to approximate socially fair k-medians by a factor better than $\Omega(\log \ell / \log \log \ell)$ unless NP $\subseteq \cap_{\delta > 0} \text{DTIME}(2^{n^\delta})$. This hardness result holds even for uniform and line metrics. We remark that since the result by Bhattacharya et al. (2014) holds for uniform metrics – metrics where all distances are either 0 or 1 – the same $\Omega(\log \ell / \log \log \ell)$-hardness result also applies to socially fair clustering with other values of $p > 1$.

Our Results. In this paper, we consider socially fair clustering for arbitrary $p \in [1, \infty)$ and arbitrary demands/weights. We do not require groups P_j to be disjoint.

Our main contribution is an $(e^{O(p)} \cdot \frac{\log \ell}{\log \log \ell})$-approximation algorithm for socially fair ℓ_p-clustering problem. For socially fair k-means, our algorithm improves the $O(\ell)$-approximation algorithms by Abbasi et al. and Ghadiri et al. to $O(\frac{\log \ell}{\log \log \ell})$. For socially fair k-medians, our algorithm improves the $O(\log n + \log \ell)$-approximation algorithm by Anthony et al. to $O(\frac{\log \ell}{\log \log \ell})$. The hardness result by Bhattacharya et al. (2014) shows that our approximation guarantee for socially fair k-clustering with $p = O(1)$, which includes k-medians and k-means, is optimal up to a constant factor unless NP $\subseteq \cap_{\delta > 0} \text{DTIME}(2^{n^\delta})$. We also give an $(e^{O(p)}/\gamma, 1/(1 - \gamma))$-bicriteria approximation for the socially fair ℓ_p-clustering problem (where $\gamma \in (0, 1)$). This result generalizes the bicriteria guarantee of Abbasi et al. (2021) for the k-means and k-medians objectives to the case of arbitrary $p \geq 1$ and arbitrary demands w_j.

---

1. Abbasi et al. call this problem *fairness under equitable group representation*.

Theorem 2 *There exists a polynomial-time algorithm that computes an $(e^{O(p)} \frac{\log \ell}{\log \log \ell})$-approximation for the socially fair ℓ_p-clustering problem (where ℓ is the number of groups).*

Theorem 3 *There exists an algorithm that computes a bicriteria $(e^{O(p)}/\gamma, 1/(1-\gamma))$-approximation for the socially fair ℓ_p-clustering problem (where $\gamma \in (0, 1)$). The algorithm finds a solution with at most $k/(1 - \gamma)$ centers, whose cost is at most $e^{O(p)}/\gamma$ times the optimal cost for k centers.*

Below, we will mostly focus on proving Theorem 2. We prove Theorem 3 in Section 6. Our algorithms are based on linear programming. However, as shown by Abbasi et al., the integrality gap of the natural LP relaxation for the socially fair ℓ_p-clustering is $\Omega(\ell)$. In order to get an approximation factor better than $\Theta(\ell)$, we strengthen the LP by introducing an extra set of constraints. Loosely speaking, new constraints require that each point u be connected only to centers at distance at most Δ from u, where the value of Δ depends on the specific point u (we discuss these constraints in detail below). Once we solve the LP relaxation with additional constraints, we apply the framework developed by Charikar, Guha, Tardos, and Shmoys (2002) for the k-medians problem. Using their framework, we transform the problem instance and the LP solution. We get an instance with a set of points P' and an LP solution that "fractionally" opens a center at every point $u \in P'$; specifically, each point $u \in P'$ will be at least $(1 - \gamma)$-fractionally open where $\gamma = 1/10$. Now to obtain our approximation results, we use independent sampling in combination with some techniques from Charikar et al. (2002). The analysis crucially uses the LP constraints we introduced. Our bicriteria approximation algorithm simply outputs set $C = P'$.

Related Work. Clustering has been an active area of research in the domain of fairness for unsupervised learning. One notion of group fairness for clustering, introduced by Chierichetti et al. (2017), requires that output clusters are balanced. This notion of fairness has been extended in a series of papers (Abraham et al., 2019; Bercea et al., 2019; Bera et al., 2019; Schmidt et al., 2019; Backurs et al., 2019; Ahmadian et al., 2019; Huang et al., 2019; Abraham et al., 2019). Another well-studied notion of group fairness for clustering requires the chosen centers fairly represent the underlying population. Various aspects of clustering under this notion of fairness have been studied in the literature (Hajiaghayi et al., 2010; Krishnaswamy et al., 2011; Chen et al., 2016; Krishnaswamy et al., 2018; Kleindessner et al., 2019; Chiplunkar et al., 2020; Jones et al., 2020). There has been also extensive research on other notions of fairness for clustering (Chen et al., 2019; Jung et al., 2020; Mahabadi and Vakilian, 2020; Micha and Shah, 2020; Kleindessner et al., 2020; Brubach et al., 2020; Anderson et al., 2020).

2. Preliminaries

We denote the distance from point u to set C by

$$d(u, C) = \min_{v \in C} d(u, v).$$

To simplify notation, we will assume below that weights w_j are defined on the entire set P, but $w_j(u) = 0$ for $u \notin P_j$. We denote $w(v) = \sum_{j \in [\ell]} w_j(v)$. We will use the following definitions and results in the paper.

Definition 4 (approximate triangle inequality) *A distance function satisfies the α-approximate triangle inequality over a set of points P if, $\forall u, v, w \in P, d(u, w) \leq \alpha \cdot (d(u, v) + d(v, w))$*

Claim 5 (Corollary A.1 in (Makarychev et al., 2019)) *Let (P, d) be a metric space. Consider distance function $d(u, v)^p$. It satisfies the α_p-approximate triangle inequality for $\alpha_p = 2^{p-1}$.*

Theorem 6 (Bennett's Inequality (Bennett, 1962); also see Theorem 2.9.2 in (Vershynin, 2018)) *Let $X_1, \ldots, X_n$ be independent mean-zero random variables and $S = \sum_{i=1}^{n} X_i$. Assume that (a) $\mathbf{Var}[S] \leq \sigma^2$ and (b) for all $i \in [n]$, $|X_i| \leq M$ always. Then, for every $t \geq 0$, we have*

$$\Pr[S \geq t] \leq \exp\left(-\frac{\sigma^2}{M^2} h\left(\frac{tM}{\sigma^2}\right)\right) \leq \exp\left(-\frac{t}{2M} \log\left(\frac{tM}{\sigma^2} + 1\right)\right),$$

for $h(x) = (1 + x)\ln(1 + x) - x^2 \geq \frac{1}{2} \cdot x \log(x + 1)$.

3. LP Relaxations for Socially Fair ℓ_p-Clustering

In this section, we describe an LP-relaxation for the socially fair ℓ_p-clustering. We start with a natural LP relaxation for the problem studied by Abbasi et al. (2021). The relaxation is a generalization of the standard LP for k-means and k-medians clustering (Charikar et al., 2002): For every $v \in P$, we have an LP variable y_v that denotes whether v belongs to the set of selected centers (in an integral solution, $y_v = 1$ if $v \in C$ and 0 otherwise); for every $u, v \in P$, we have an LP variable x_{uv} that denotes whether v is the closest center to u in the selected set of centers C.

LP Relaxation: $BasicClusterLP(\{w_j\})$

$$\text{minimize} \quad \max_{j \in [\ell]} \sum_{u \in P_j, v \in P} w_j(u) \cdot d(u, v)^p \cdot x_{uv}$$

$$\text{s.t.} \quad \sum_{v \in P} x_{uv} = 1 \qquad\qquad \forall u \in P$$

$$\sum_{v \in P} y_v \leq k$$

$$x_{uv} \leq y_v \qquad\qquad \forall u, v \in P$$

$$x_{uv}, y_u \geq 0 \qquad\qquad \forall u, v \in P$$

Note that the objective is not linear as written. However, we can rewrite this relaxation as a true LP by introducing a new variable A, adding LP constrains $A \geq \sum_{u \in P_j, v \in P} w_j(u) \cdot d(u, v)^p \cdot x_{uv}$ for each j, and then minimizing A in the objective.

As was shown by Abbasi et al., this LP relaxation has an integrality gap of $\Omega(\ell)$ (for all $p \in [1, \infty)$). As discussed in the Introduction, we strengthen this LP by introducing an extra set of constraints. To describe these constraints, we need some notation. For each point $v \in P$, we denote the ball of radius $r \geq 0$ around v by $\mathcal{B}(v, r) := \{u \in P \mid d(v, u) \leq r\}$. We define the volume of a ball $\mathcal{B}(v, r)$ as $\text{vol}_v(r) = \max_{j \in [\ell]} \sum_{u \in \mathcal{B}(v,r)} w_j(u) \cdot r^p$.

Remark 7 *We use this definition of $\text{vol}_v(r)$ so as to ensure that the following property holds. Consider a set of centers C and assume that the distance from all points in $\mathcal{B}(v, r)$ to C is at least r. Then the cost of solution C is at least $\text{vol}_v(r)$.*

Clearly, function $\text{vol}_v(r)$ is a non-decreasing function of r that goes to infinity as $r \to \infty$ (unless all demands w_j are identically 0). Further, if there are no points at distance exactly r from v,

then $\mathrm{vol}_v(r)$ is continuous at r; otherwise, it is right-continuous at r and may or may not be left-continuous. Denote $\mathrm{vol}_v(r-0) = \lim_{s \to r-0} \mathrm{vol}_v(s)$ (the limit of vol_v at r from the left); we let $\mathrm{vol}_v(0-0) = 0$. Given a *budget* parameter z, define $\Delta_z(v) = \min\{r : \mathrm{vol}_v(r) \geq z\}$. Note that $\mathrm{vol}_v(\Delta_z(v) - 0) \leq z \leq \mathrm{vol}_v(\Delta_z(v))$. Claim 8 explains why we consider $\Delta_z(v)$.

Claim 8 *Consider an instance of the socially fair ℓ_p-clustering problem. Denote its optimal cost by z^*. Let C^* be an optimal set of centers and $z \geq z^*$. Then for all $v \in P$ with $w(v) = \sum_{j \in [\ell]} w_j(v) > 0$, we have $d(v, C^*) \leq 2\Delta_z(v)$.*

Proof Denote $R = \Delta_z(v)$. Assume to the contrary that there exists a point v such that $d(v, C^*) > 2R$. Then, for every $u \in \mathcal{B}(v, R)$, we have $d(u, C^*) \geq d(v, C^*) - d(u, v) > R$.

If $z > 0$, choose j so that $\sum_{u \in \mathcal{B}(v,R)} w_j(u) \cdot R^p = \mathrm{vol}_v(R) \geq z$. If $z = 0$, choose j so that $w_j(v) > 0$. Note that in either case, $S := \sum_{u \in \mathcal{B}(v,R)} w_j(u) > 0$. We have,

$$\text{fair-cost}(C^*) \geq \text{cost}(C^*, w_j) \geq \sum_{u \in \mathcal{B}(v,R)} w_j(u) d(u, C^*)^p$$

$$\overset{\text{since } S > 0}{>} \sum_{u \in \mathcal{B}(v,R)} w_j(u) R^p \geq z \geq z^* = \text{fair-cost}(C^*).$$

We get a contradiction. ∎

We now state our strengthened LP relaxation, which has a new family of constraints (5).

LP Relaxation: $ClusterLP(\{w_j\}, z, \lambda)$

$$\text{minimize } \max_{j \in [\ell]} \sum_{u \in P_j, v \in P} w_j(u) \cdot d(u, v)^p \cdot x_{uv}$$

$$\text{s.t.} \qquad \sum_{v \in P} x_{uv} = 1 \qquad\qquad \forall u \in P \tag{2}$$

$$\sum_{v \in P} y_v \leq k \tag{3}$$

$$x_{uv} \leq y_v \qquad\qquad \forall u, v \in P \tag{4}$$

$$x_{vu} = 0 \qquad\qquad \forall u, v \text{ s.t. } w(v) > 0 \text{ and } d(u, v) > \lambda \Delta_z(v) \tag{5}$$

$$x_{uv}, y_u \geq 0 \qquad\qquad \forall u, v \in P \tag{6}$$

It follows from Claim 8 that this is a valid relaxation if $\lambda \geq 2$ and z is at least z^* (the cost of the optimal fair clustering). We remark that the bicriteria approximation can be achieved from rounding $BasicClusterLP$; however, for the simplicity of exposition, we also use $ClusterLP$ in our bicriteria approximation algorithm.

We show that the integrality gap of $ClusterLP(\{w_j\}, z, \lambda)$ is $\Omega(\log \ell / \log \log \ell)$ and thus our rounding procedure is essentially optimal.

Theorem 9 *The integrality gap of $ClusterLP(\{w_j\}, z, \lambda)$ is $\Omega(\log \ell / \log \log \ell)$.*

We make the statement of Theorem 9 precise and prove it in Appendix C.

Outline of Our Algorithms. We provide two approximation guarantees for fair ℓ_p-clustering. The first algorithm gives an $(e^{O(p)}\frac{\log \ell}{\log \log \ell})$-approximation and the second one gives a bicriteria $(e^{O(p)}/\gamma, 1/(1-\gamma))$-approximation.

We will assume below that we have an approximation z_g for the cost z^* of the optimal solution such that $z^* \leq z_g \leq 2z^*$. A standard argument shows that we can do that: we can simply run our algorithm with various values of z_g and, loosely speaking, output the best clustering the algorithm finds. We formally explain why we can assume that we have such a value of z_g in Appendix B.

Our algorithms consist of two steps. In the first step, which is common for both our algorithms, we construct a $(1-\gamma)$-restricted solution of $ClusterLP$, defined as follows.

Definition 10 *A solution (x, y) of $ClusterLP(\{w_j\}, z, \lambda)$ is called $(1 - \gamma)$-restricted w.r.t. $P' \subseteq P$ if $y_u \geq 1 - \gamma$ for all $u \in P'$ and $y'_u = 0$ for all $u \notin P'$.*

We find a $(1 - \gamma)$-restricted solution by closely following the approach by Charikar, Guha, Tardos, and Shmoys (2002). However, as we work with a different objective function and LP, some careful modifications to the approach by Charikar et al. are required. The second steps for our approximation and bicriteria approximation algorithms are different. The former uses an independent randomized rounding in combination with some techniques from Charikar et al.; the latter algorithm does not actually need any rounding – it simply outputs $C = P'$.

4. Step 1: Constructing $(1 - \gamma)$-Restricted Solutions

4.1. Overview

In this section, we present a polynomial-time algorithm that given an instance $\mathcal{I}$ with demand functions $\{w_j\}$ constructs an instance $\mathcal{I}'$ with new demands $\{w'_j\}$ on the same set of points P. Let P' be the support of $\{w'_j\}$: $P' = \{u : w'_j(u) \neq 0 \text{ for some } j \in [\ell]\}$. The new instance will satisfy the following properties (which we now state informally).

1. All points in P' are well separated (the distance between every two points in P' is "large").

2. Let (x, y) be an optimal LP solution for $ClusterLP(\{w_j\}, z_g, 2)$. The LP cost of (x, y) w.r.t. new demands $\{w'_j\}$ is at most that w.r.t. the original demands $\{w_j\}$.

3. The cost of every solution C w.r.t. original demands $\{w_j\}$ is at most a constant factor greater than the cost of C w.r.t. demands $\{w'_j\}$ (for every fixed p).

After we show how to transform $\mathcal{I}$ to $\mathcal{I}'$ and prove that $\mathcal{I}'$ satisfies properties (1)-(3), we describe how to convert the optimal LP solution (x, y) for $\mathcal{I}$ to a solution (x', y') for $\mathcal{I}'$, which is $(1 - \gamma)$-restricted w.r.t. P' (see Definition 10).

No we observe that it is sufficient to design a "good" rounding scheme only for $(1-\gamma)$-restricted LP solutions: we use the transformations discussed above, then apply the rounding scheme to the LP solution (x', y') for $\mathcal{I}'$, obtain a "good" solution C for $\mathcal{I}'$, and then output C as a solution for $\mathcal{I}$.

The proofs of all claims and lemmas from Sections 4.2 and 4.3 appear in Appendix A.

4.2. Consolidating Locations

In this subsection, we describe how we transform instance $\mathcal{I}$ to instance $\mathcal{I}'$. We call this step location consolidation. Let (x, y) be an optimal LP solution for $ClusterLP(\{w_j\}, z_{\boldsymbol{g}}, 2)$ where $z_{\boldsymbol{g}} \in [z^*, 2z^*]$ and z^* is the cost for the optimal integral solution of the fair clustering instance. To describe the location consolidation algorithm, we need to define a notion of "fractional distance of point u to the center" according to (x, y). We define the fractional distance $\mathcal{R}(u)$ for $u \in P$ as follows:

$$\mathcal{R}(u) := \left(\sum_{v \in P} d(u, v)^p \cdot x_{uv} \right)^{1/p} \tag{7}$$

Note that if (x, y) is an integral solution, then $\mathcal{R}(u)$ is simply the distance from u to the center of the cluster u is assigned to.

Claim 11 *For each group $j \in [\ell]$, we have $\sum_{u \in P} w_j(u) \cdot \mathcal{R}(u)^p \le z^*$.*

Algorithm 1 performs location consolidation. After we initialize variables (lines 2–3), we sort

---

Algorithm 1 Consolidating locations.

1: **Input:** (x, y) is an optimal solution of $ClusterLP(\{w_j\}, \Delta, \gamma)$
2: $\mathcal{R}(v) = \left(\sum_{u \in P} d(v, u)^p \cdot x_{vu} \right)^{1/p}$ **for all** $v \in P$
3: $w'_j(v) = w_j(v)$ **for all** $v \in P$ and $j \in [\ell]$
4: **sort** the points in P so that $\mathcal{R}(v_1) \le \mathcal{R}(v_2) \le \cdots \le \mathcal{R}(v_n)$
5: **for** $i = 1$ to $n - 1$ **do**
6:     **for** $j = i + 1$ to n **do**
7:         **if** $d(v_i, v_j) \le \frac{2}{\gamma^{1/p}} \mathcal{R}(v_j)$ and $\sum_{c \in [\ell]} w'_c(v_i) > 0$ **then**
8:             $w'_t(v_i) = w'_t(v_i) + w'_t(v_j)$ **for all** $t \in [\ell]$
9:             $w'_t(v_j) = 0$ **for all** $t \in [\ell]$
10:         **end if**
11:     **end for**
12: **end for**

---

all points in P according to their fractional distance in a non-decreasing order $- v_1, \cdots, v_n -$ so that $\mathcal{R}(v_1) \le \mathcal{R}(v_2) \le \cdots \le \mathcal{R}(v_n)$ (line 4). Then, we consider the points in this order one-by-one. When processing a point v_i with non-zero demand, we check whether there exists another point v_j with non-zero demand such that $j > i$ and $d(v_i, v_j) \le \frac{2}{\gamma^{1/p}} \mathcal{R}(v_j)$ (line 7). If there is such a point, we add the demands of v_j to v_i and set the demands of v_j to zero (lines 8-9). When the algorithm runs the described procedure, we say that it *moves* the demand of v_j to v_i. Note that after the algorithm processes v_i, it never moves the demand of v_i to another point. Thus, the demand of each point can be moved at most once.

We run Algorithm 1 on optimal LP solution (x, y) and obtain a new set of demands $\{w'_j\}$. Let $w'(v) = \sum_{j \in [\ell]} w'_j(v)$, and P' be the support of w'. Claim 12 shows that all points in P' are well-separated.

Claim 12 *For every pair of $u, v \in P'$, $d(u, v) > \frac{2}{\gamma^{1/p}} \cdot \max(\mathcal{R}(u), \mathcal{R}(v))$.*

Finally, we show that every solution (x, y) for the original LP relaxation is also a feasible solution with the same or smaller cost for the LP with new demands $\{w'_j\}$.

Lemma 13 *Let (x, y) be a feasible solution for $ClusterLP(\{w_j\}, z, \lambda)$ with cost z'. Then (x, y) is a feasible solution for $ClusterLP(\{w'_j\}, z, \lambda)$ with cost at most z'.*

4.3. Consolidating Centers

In the previous section, we constructed an instance with a well-separated set of points P' that have positive demands $\{w'_j\}$. In this section, we simplify the structure of the set of "opened centers" in the LP solution – points $u \in P$ with $y_u > 0$. Note that y_u may be positive even if $u \notin P'$. We will transform the LP solution (x, y) and obtain a new solution (x', y') with approximately the same LP cost such that $y'_u > 0$ only if $u \in P'$. We will see then that (x', y') is a $(1 - \gamma)$-restricted solution of $ClusterLP$.

Our approach is identical to that in Charikar et al. (2002). If $v \in P \setminus P'$ and $y_v > 0$, we move center v to the closest to v point v' in P' by letting $y'_{v'} = y'_{v'} + y_v$. Then we close the center v by letting $y'_v = 0$. If $v \in P'$, we keep the center at v. See Algorithm 2 for the formal description of this procedure.

Algorithm 2 Consolidating centers.

1: **Input:** P', x, y
2: $x' = x, y' = y$
3: **for all** $v \in P \setminus P'$ with $y'_v > 0$ **do**
4:    let v' be a closest to v point in P'.
5:    $y'_{v'} = \min(1, y'_{v'} + y'_v), y'_v = 0$
6:    $x'_{uv'} = x'_{uv'} + x'_{uv}$ and $x'_{uv} = 0$ **for all** $u \in P'$
7: **end for**

Consider a point $u \in P'$ that is fractionally served by center $v \notin P'$ in LP solution (x, y); that is, $x_{uv} > 0$ and $v \notin P$. In the new solution (x', y'), it is served by center v'. We show that the distance from u to the new center is greater than that to the old one by at most a factor of 2.

Claim 14 *Consider a point $v \in P \setminus P'$ and let v' be the nearest neighbor of v in the set P'. Then, for every $u \in P'$, $d(u, v') \leq 2d(u, v)$.*

Lemma 15 *Algorithm 2, given a solution (x, y) for $ClusterLP(\{w'_j\}, z, \lambda)$ of cost z', constructs a $(1 - \gamma)$-restricted w.r.t. P' solution (x', y') for $ClusterLP(\{w'_j\}, z, 2\lambda)$ of cost at most $2^p z'$.*

We run Algorithm 2 on the optimal solution (x, y) for $ClusterLP(\{w_j\}, z_{\boldsymbol{g}}, 2)$ and obtain a feasible $(1-\gamma)$-restricted solution (x', y') for $ClusterLP(\{w_j\}, z_{\boldsymbol{g}}, 4)$. Since $ClusterLP(\{w_j\}, z_{\boldsymbol{g}}, 2)$ is a relaxation for the fair ℓ_p-clustering, the cost of (x, y) w.r.t. demands $\{w_j\}$ is at most z^*. By Lemma 13, the cost of (x, y) w.r.t $\{w'_j\}$ is also at most z^*. Finally, by Lemma 15, the cost of (x', y') is at most $2^p z^*$

4.4. Relating the solution costs w.r.t. $\mathcal{I}'$ and $\mathcal{I}$

In the previous sections, we transformed given instance $\mathcal{I}$ to instance $\mathcal{I}'$ with demands $\{w'_j\}$ and well-separated locations P', then converted an optimal LP solution (x, y) to a $(1 - \gamma)$-restricted solution (x', y'). Now we upper bound the cost of every solution C w.r.t. original demands $\{w_j\}$ in terms of the cost w.r.t. demands $\{w'_j\}$.

Lemma 16 *Let z^* be the cost of the optimal solution for $\mathcal{I}$. Assume that $z_{\boldsymbol{g}} \in [z^*, 2z^*]$. For any integral solution C' with* $\text{fair-cost}(C', \{w'_j\}) \le z'$,

$$\text{fair-cost}(C', \{w_j\}) \le \frac{2^{2p-1}}{\gamma} \cdot z^* + 2^{p-1} \cdot z'.$$

Proof Consider the execution of Algorithm 1. The algorithm may either move the demand of point u to some other point v or keep it at u. Let $u' = v$ in the former case and $u' = u$ in the latter case (for every $u \in P$). Note that in either case $d(u, u') \le \frac{2}{\gamma^{1/p}} \mathcal{R}(u)$. Therefore,

$$
\begin{aligned}
d(u, C')^p &\le \alpha_p \cdot (d(u, u')^p + d(u', C')^p) && \triangleright \text{ Claim 5 (approximate triangle inequality)} \\
&\le \alpha_p \cdot \left(\frac{2^p}{\gamma} \cdot \mathcal{R}(u)^p + d(u', C')^p \right) && \triangleright \text{ we upper bound } d(u, u') && (8)
\end{aligned}
$$

where $\alpha_p = 2^{p-1}$. Hence, for each group j,

$$
\begin{aligned}
\sum_{u \in P} w_j(u) \cdot d(u, C')^p &\le \sum_{u \in P} w_j(u) \cdot \left(\frac{2^{2p-1}}{\gamma} \cdot \mathcal{R}(u)^p + 2^{p-1} \cdot d(u', C')^p \right) && \triangleright \text{ by (8)} \\
&\le \frac{2^{2p-1}}{\gamma} \cdot z^* + 2^{p-1} \cdot \sum_{u \in P} w_j(u) \cdot d(u', C')^p && \triangleright \text{ Claim 11} \\
&= \frac{2^{2p-1}}{\gamma} \cdot z^* + 2^{p-1} \cdot \sum_{v \in P'} w'_j(v) \cdot d(v, C')^p \le \frac{2^{2p-1}}{\gamma} \cdot z^* + 2^{p-1} \cdot z'.
\end{aligned}
$$

It follows that $\text{fair-cost}(C', \{w_j\}) \le \frac{2^{2p-1}}{\gamma} \cdot z^* + 2^{p-1} \cdot z'$. ∎

5. Step 2: Rounding $(1 - \gamma)$-Restricted Solutions

5.1. Randomized Rounding for Multiplicative Approximation

Before describing our randomized rounding procedure, we prove several lemmata, which we will use in the analysis of the algorithm.

Lemma 17 *Consider an instance of socially fair clustering. Let $\{w'_j\}$ be the set of weights computed by Algorithm 1 and (x', y') be a feasible solution for $ClusterLP(\{w'_j\}, z_{\boldsymbol{g}}, 4)$ computed by Algorithm 2. Consider some group j and some point $v \in P'$ with $x'_{vv} < 1$. Let v' be a point closest to v in P' other than v itself. Then,*

$$w'_j(v) \cdot d(v, v')^p \le \left(2 \cdot 4^p + \frac{8^p}{\gamma} \right) z^*,$$

where z^* is the cost of the optimal solution and $z_{\boldsymbol{g}} \in [z^*, 2z^*]$.

9

Proof We assume that $w'_j(v) > 0$, as otherwise the statement is trivial. Let $Q \subseteq P$ be the set of points whose demands have moved to v by Algorithm 1. Further, let $Q_1 = Q \cap \mathcal{B}(v, \Delta_{z_g}(v) - 0) := \{u : d(u, v) < \Delta_{z_g}(v)\}$ and $Q_2 = Q \setminus Q_1$. Note that

$$w'_j(v) = \sum_{u \in Q_1} w_j(u) + \sum_{u \in Q_2} w_j(u).$$

We first get an upper bound on $d(v, v')$. Since $x'_{vv} < 1$, we know that $x'_{vu} > 0$ for some $u \in P'$ other than v. Constraint (5) implies that $d(v, u) \leq 4\Delta_{z_g}(v)$ (here we use that (x', y') is a feasible solution for $ClusterLP(\{w'_j\}, z_g, 4)$). Now recall that v' is a closest point to v in P' other than v. Thus, $d(v, v') \leq d(v, u) \leq 4\Delta_{z_g}(v)$. In particular, $d(v, v') \leq 4d(v, q)$ for points $q \in Q_2$. Further, Algorithm 1 moves demand from q to v, only if $d(v, q) \leq \frac{2}{\gamma^{1/p}}\mathcal{R}(q)$. For $q \in Q_2$, we get $d(v, v') \leq 4d(v, q) \leq \frac{8}{\gamma^{1/p}}\mathcal{R}(q)$. Therefore,

$$w'_j(v) \cdot d(v, v')^p = \sum_{u \in Q_1} w_j(u)d(v, v')^p + \sum_{u \in Q_2} w_j(u)d(v, v')^p$$

$$\leq \sum_{u \in Q_1} w_j(u)(4\Delta_{z_g}(v))^p + \frac{8^p}{\gamma} \sum_{u \in Q_2} w_j(u)\mathcal{R}(u)^p$$

Note that $\sum_{u \in Q_1} w_j(u)\Delta_{z_g}(v)^p \leq \text{vol}_v(\Delta_{z_g} - 0) \leq z_g$ by the definition of Δ_{z_g}. Using this inequality and Claim 11, we get

$$w'_j(v) \cdot d(v, v')^p \leq 4^p z_g + \frac{8^p}{\gamma} z^*.$$

The statement of the lemma follows, since we assume that $z_g \leq 2z^*$. ∎

Let us now define a forest $F = (P', E)$ on P'. We sort all pairs $\{u, v\}$ of distinct points in P' according to the distance between them in ascending order (breaking ties arbitrarily). For every point $u \in P'$, we choose the first pair $\{u, v\}$ it appears in and let $u' = v$. Then, u' is a closest point to u in P' other than u itself. For every u, we add edge (u, u') to our graph F (we add every edge at most once). It is easy to see that the obtained graph is a forest.

Lemma 18 (cf. Charikar et al. (2002)) *Let (x', y') be a feasible $(1 - \gamma)$-restricted solution returned by Algorithm 2 with $\gamma < 1/2$. There exists a feasible solution (x'', y') of cost at most that of (x', y') such that the following holds. For every $v \in P'$, we have $x''_{vv} = y'_v \geq 1 - \gamma$, $x''_{vv'} = 1 - x''_{vv}$ and $x''_{vu} = 0$ for $u \neq \{v, v'\}$, where v' is as in the definition of F.*

Proof We simply let $x''_{vv} = 1 - y'_v$ for every $v \in P'$, $x''_{vv'} = 1 - y'_v$, and $x''_{vu} = 0$ for $u \notin \{v, v'\}$. Since (x, y) is a $(1 - \gamma)$-restricted solution and $\gamma < 1/2$, $x''_{vu} = 1 - y'_v \leq 1 - (1 - \gamma) = \gamma \leq 1 - \gamma \leq y'_u$, as required. It is easy to see that all other LP constraints are also satisfied. Further, we chose x'' in an optimal way for the given y'. Thus, the cost of (x'', y') is at most that of (x', y'). ∎

Fix $\gamma = 1/10$. For each $u \in P'$, let $p_u = (1 - y'_v)/\gamma$. Note that $0 \leq p_u \leq 1$ since $0 \leq 1 - y'_v \leq \gamma$. We have

$$\sum_{u \in P'} p_u = \gamma^{-1} \sum_{u \in P'} (1 - y'_u) = \gamma^{-1} \left(|P'| - \sum_{u \in P'} y'_u \right) \geq \gamma^{-1}(|P'| - k).$$

For every tree T in the forest F, do the following. Choose an arbitrary root r in T. Partition T into layers based on their depth, starting with the root. Let A_T be the union of every other layer; that is, A_T is the set of vertices of even depth. Finally, let A be the union of all sets A_T over T in F. Note that all neighbors of $u \in A$ are not in A; all neighbors of $u \notin A$ are in A.

If $\sum_{u \in A} p_u \geq \frac{1}{2\gamma}(|P'| - k)$, let $S = A$; otherwise, let $S = P' \setminus A$. In either case, $\sum_{u \in S} p_u \geq \frac{1}{2\gamma}(|P'| - k)$. Finally, we construct our combinatorial solution C:

- if $|P'| \leq k$, let $C = |P'|$; otherwise, proceed as follows

- add all points from $P' \setminus S$ to C,

- add each point $v \in S$ to C with probability $1 - p_v$ (independently).

We show now that $|C| \leq k$ with probability at least $3/4$ and fair-cost$(C) \leq O\left(2^{3p}\right) \frac{\log \ell}{\log \log \ell} \cdot z^*$ (where z^* is the cost of the optimal solution) with probability at least $1 - 1/(2\ell)$.

Theorem 19 *With probability at least* $1/4$, $|C| \leq k$ *and* fair-cost$(C, \{w'_j\}) = O(2^{3p} \cdot \frac{\log \ell}{\log \log \ell} z^*)$.

Proof If $|P'| \leq k$, then $|C| = |P'| \leq k$ and the cost of C w.r.t. demands $\{w'_j\}$ is 0; thus, the statement of the theorem trivially holds. We assume below that $|P'| > k$. By Lemmata 15 and 18, (x', y') and (x'', y') are feasible solutions for $ClusterLP(\{w'_j\}, z_g, 4)$. Let z be the LP cost of (x'', y') w.r.t. demands $\{w'_j\}$. Then $z \leq 2^p z^*$.

First, we bound the size of C. Let X_v be the indicator random variable of the event $v \notin C$; i.e., $X_v = 1$ if $v \notin C$ and 0 otherwise. Define $X = \sum_{v \in P'} X_v$. Now we are lower bounding the number X of points in P' that are *not* centers in C.

$$\mathbf{E}[X] = \mathbf{E}\left[\sum_{v \in P'} X_v\right] = \sum_{v \in S} \mathbf{E}[X_v] + \sum_{v \in P' \setminus S} \mathbf{E}[X_v] = \sum_{v \in S} p_v \geq \frac{|P'| - k}{2\gamma}.$$

Applying the Chernoff bound, we get for $\varepsilon = 1 - 2\gamma$,

$$\Pr(X \leq |P'| - k) \leq \Pr(X \leq (1 - \varepsilon)\mathbf{E}(X)) \leq \exp(-\varepsilon^2 \mathbf{E}[X]/2)$$

Note that $\mathbf{E}[X] \geq \frac{1}{2\gamma} = 5$ and $\varepsilon \geq 4/5$, since $|P'| \geq k + 1$ and $\gamma = 1/10$. Thus, $\Pr[X \leq (1 - \varepsilon)\mathbf{E}(X)] < 1/4$. Hence, with probability at least $3/4$,

$$|C| \leq |P'| - X \leq |P'| - (|P'| - k) = k. \tag{9}$$

Fix $j \in [\ell]$. Now we show that $\text{cost}(C, w'_j)$ is at most $O\left(2^{3p-1} \log \ell / \log \log \ell\right) z^*$ with high probability. Then applying the union bound, we will get the same bound for all $j \in [\ell]$ and thus for fair-cost$(C, \{w'_j\})$.

For every $v \in P'$, let $Y_v = w'_j(v) \cdot d(v, C)^p$. Note that for every $v \in P'$ either $v \notin S$ or $v' \notin S$ (since v and v' are neighbors in F). Therefore, we always have that at least one of the points v and v' is in C. Further, if $X_v = 0$, then $v \in C$, and thus $d(v, C) = 0$; if $X_v = 1$, then $d(v, C) = d(v, v')$. Hence, $Y_v = w'_j(v) d(v, v')^p X_v$. From Lemma 17, we get for each $v \in P'$,

$$Y_v \leq w'_j(v) \cdot d(v, v')^p \leq \left(2 \cdot 2^{2p} + \frac{2^{3p}}{\gamma}\right) z^* = O\left(2^{3p} z^*\right) \quad \text{(always)}.$$

Let $Z_j := \sum_{v \in P_j} Y_v = \sum_{v \in S} Y_v$ be the cost of group j w.r.t. weights $\{w'_j\}$. Note that all random variables $\{Y_v\}_{v \in P_j}$ are independent. Random variables Y_u for $u \notin S$ are identically equal to 0. We have,

$$\mathbf{E}[Z_j] \leq \sum_{v \in S} w'_j(v) d(v, v')^p p_v = \frac{1}{\gamma} \sum_{v \in S} w'_j(v) d(v, v')^p \underbrace{(1 - y'_v)}_{x''_{vv'}} \leq \frac{z}{\gamma} \leq \frac{2^p}{\gamma} z^*$$

(recall that z is the cost of the LP solution (x'', y')) and

$$\mathbf{Var}[Z_j] = \sum_{v \in S} \mathbf{Var}[Y_v] \leq \sum_{v \in S} (w'_j(v) d(v, v')^p)^2 p_v \overset{\text{by Lemma 17}}{\leq} O\left(2^{3p} z^*\right) \cdot \mathbf{E}[Z_j] = O\left(2^{4p} (z^*)^2\right).$$

Now we will use Bennett's inequality (Theorem 6) to bound random variables Y_j. To do so, we define *zero-mean* versions of random variables Y_v and Z_j. Let $Y'_v := Y_v - \mathbf{E}[Y_v]$ and $Z'_j := \sum_{v \in P'_j} Y'_v$. Note that, $|Y'_v| = O(2^{3p} z^*)$ (always) and $\mathbf{Var}[Z'_j] = \mathbf{Var}[Z_j] = O\left(2^{4p} (z^*)^2\right)$. Applying Bennett's inequality to Z'_j, we get for $\tau > 0$

$$\Pr\left(Z'_j \geq \tau z^*\right) \leq \exp\left(-\Omega\left(\frac{\tau}{2^{3p}}\right) \log\left(\Omega\left(\frac{\tau}{2^p}\right) + 1\right)\right)$$

Letting $\tau = c \cdot 2^{3p} \frac{\log \ell}{\log \log \ell}$ for large enough c, we get that $\Pr\left(Z'_j \geq \tau z^*\right) \leq \frac{1}{2\ell^2}$, which implies that $\Pr\left(Z_j \geq c' \cdot 2^{3p} \frac{\log \ell}{\log \log \ell} z^*\right) \leq \frac{1}{2\ell^2}$ for some absolute constant c'. Applying the union bound, we get that with probability at least $1 - \frac{1}{2\ell}$, the following upper bound on the fair cost holds

$$\text{fair-cost}(C, \{w'_j\}) = \max_{j \in [\ell]} \sum_{v \in P'} w'_j(v) \cdot d(v, C)^p = O\left(2^{3p} \cdot \frac{\log \ell}{\log \log \ell} z^*\right). \tag{10}$$

We conclude that with probability at least $1 - \frac{1}{2\ell} - 1/4 \geq \frac{1}{4}$, both $|C| \leq k$ and (10) holds. ∎

Algorithm 3 ($e^{O(p)} \frac{\log \ell}{\log \log \ell}$)-approximation algorithm for socially fair ℓ_p-clustering

---

1: **Input:** An instance $\{w_j\}$ of socially fair clustering, $z_{\boldsymbol{g}} \in [z^*, 2z^*]$, desired error probability ε
2: $\gamma = 1/10, \lambda = 2$
3: Solve $ClusterLP(\{w_j\}, z_{\boldsymbol{g}}, 2)$. Let (x, y) be an optimal fractional solution.
4: Run Algorithm 1 and obtain new demands $(\{w'_j\})$ and set P'
5: Run Algorithm 2 and obtain LP solution (x', y')
6: Define x'' as in Lemma 18.
7: Construct forest F and find set S.
8: Run randomized rounding $\lceil \log_{4/3}(1/\varepsilon) \rceil$ times
9: Let C be the best of the solutions (with at most k centers) that randomized rounding generates.
10: **return** C

Proof of Theorem 2 We put together all the steps we described in this paper. The entire algorithm is shown as Algorithm 3. As discussed in Appendix B, we may assume that $z_{\boldsymbol{g}} \in [z^*, 2z^*]$ is given to us. By Theorem 19, the probability that our randomized rounding procedure will find a feasible

solution C for demands $\{w'_j\}$ of cost at most $O\left(2^{3p}\right)\frac{\log \ell}{\log\log \ell} \cdot z^*$ is at least $1/4$. Since we run randomized rounding $\lceil \log_{4/3}(1/\varepsilon) \rceil$ times, we will succeed at least once with probability at least $1 - \varepsilon$. By Lemma 16, the cost of C w.r.t. the original demands $\{w_j\}$ is upper bounded as follows,

$$\text{fair-cost}(C, \{w_j\}) \leq \frac{2^{2p-1}}{\gamma} \cdot z^* + 2^{p-1} \cdot \text{fair-cost}(C, \{w'_j\}) = O\left(2^{4p}\right)\frac{\log \ell}{\log\log \ell} z^*$$

∎

6. Deterministic Rounding for Bicriteria Approximation

In this section we show how to get our bicriteria approximation algorithm.

Proof of Theorem 3 We solve LP relaxation $ClusterLP(\{w_j\}, z_{\boldsymbol{g}}, 2)$ and then run Algorithms 1 and 2. We obtain a fractional solution (x', y') and set P'. By Lemma 13, (x', y') is a $(1 - \gamma)$-restricted solution for $ClusterLP(\{w'_j\}, z, 4)$ w.r.t. P' of cost at most $2^p z^*$. We return set $C = P'$.

By LP constraint (3), we have $\sum_{u \in P} y_u \leq k$. Now, since (x', y') is a $(1 - \gamma)$-restricted solution, $y_u \geq 1 - \gamma$ for $u \in P'$. Therefore, $|C| = |P'| \leq k/(1 - \gamma)$, as required. Note that for every $u \in P'$, we have $d(u, C)^p = 0$. Therefore,

$$\text{fair-cost}(C, \{w'_j\}) = 0.$$

Now, we apply Lemma 16 to bound $\text{fair-cost}(C, \{w_j\})$,

$$\text{fair-cost}(C, \{w_j\}) \leq \frac{2^{2p-1}}{\gamma} \cdot z^* + 2^{p-1} \cdot \text{fair-cost}(C, \{w'_j\}) = \frac{2^{2p-1}}{\gamma} z^*.$$

∎

References

Mohsen Abbasi, Aditya Bhaskara, and Suresh Venkatasubramanian. Fair clustering via equitable group representations. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pages 504–514, 2021.

Savitha Sam Abraham, Deepak P, and Sowmya S Sundaram. Fairness in clustering with multiple sensitive attributes. *arXiv preprint arXiv:1910.05113*, 2019.

Sara Ahmadian, Alessandro Epasto, Ravi Kumar, and Mohammad Mahdian. Clustering without over-representation. In *Proceedings of the SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 267–275, 2019.

Nihesh Anderson, Suman K Bera, Syamantak Das, and Yang Liu. Distributional individual fairness in clustering. *arXiv preprint arXiv:2006.12589*, 2020.

Barbara Anthony, Vineet Goyal, Anupam Gupta, and Viswanath Nagarajan. A plant location guide for the unsure: Approximation algorithms for min-max location problems. *Mathematics of Operations Research*, 35(1):79–101, 2010.

Martin Aumüller, Rasmus Pagh, and Francesco Silvestri. Fair near neighbor search: Independent range sampling in high dimensions. In *Proceedings of the Symposium on Principles of Database Systems*, pages 191–204, 2020.

Arturs Backurs, Piotr Indyk, Krzysztof Onak, Baruch Schieber, Ali Vakilian, and Tal Wagner. Scalable fair clustering. In *Proceedings of the International Conference on Machine Learning*, pages 405–413, 2019.

George Bennett. Probability inequalities for the sum of independent random variables. *Journal of the American Statistical Association*, 57(297):33–45, 1962.

Suman Bera, Deeparnab Chakrabarty, Nicolas Flores, and Maryam Negahbani. Fair algorithms for clustering. In *Advances in Neural Information Processing Systems*, pages 4955–4966, 2019.

Ioana O Bercea, Martin Groß, Samir Khuller, Aounon Kumar, Clemens Rösner, Daniel R Schmidt, and Melanie Schmidt. On the cost of essentially fair clusterings. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*, 2019.

Sayan Bhattacharya, Parinya Chalermsook, Kurt Mehlhorn, and Adrian Neumann. New approximability results for the robust k-median problem. In *Scandinavian Workshop on Algorithm Theory*, pages 50–61. Springer, 2014.

Brian Brubach, Darshan Chakrabarti, John Dickerson, Samir Khuller, Aravind Srinivasan, and Leonidas Tsepenekas. A pairwise fair and community-preserving approach to k-center clustering. In *Proceedings of the International Conference on Machine Learning*, pages 1178–1189, 2020.

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.

Danny Z Chen, Jian Li, Hongyu Liang, and Haitao Wang. Matroid and knapsack center problems. *Algorithmica*, 75(1):27–52, 2016.

Xingyu Chen, Brandon Fain, Liang Lyu, and Kamesh Munagala. Proportionally fair clustering. In *Proceedings of the International Conference on Machine Learning*, pages 1032–1041, 2019.

Flavio Chierichetti, Ravi Kumar, Silvio Lattanzi, and Sergei Vassilvitskii. Fair clustering through fairlets. In *Advances in Neural Information Processing Systems*, pages 5036–5044, 2017.

Ashish Chiplunkar, Sagar Kale, and Sivaramakrishnan Natarajan Ramamoorthy. How to solve fair k-center in massive data models. In *Proceedings of the International Conference on Machine Learning*, pages 1877–1886, 2020.

Alexandra Chouldechova. Fair prediction with disparate impact: A study of bias in recidivism prediction instruments. *Big data*, 5(2):153–163, 2017.

Alexandra Chouldechova and Aaron Roth. The frontiers of fairness in machine learning. *arXiv preprint arXiv:1810.08810*, 2018.

Julia Dressel and Hany Farid. The accuracy, fairness, and limits of predicting recidivism. *Science advances*, 4(1), 2018.

Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Richard Zemel. Fairness through awareness. In *Proceedings of the Innovations in Theoretical Computer Science*, pages 214–226, 2012.

Michael Feldman, Sorelle A Friedler, John Moeller, Carlos Scheidegger, and Suresh Venkatasubramanian. Certifying and removing disparate impact. In *Proceedings of the International Conference on Knowledge Discovery and Data Mining*, pages 259–268, 2015.

Jorge Galindo and Pablo Tamayo. Credit risk assessment using statistical and machine learning: basic methodology and risk modeling applications. *Computational Economics*, 15(1-2):107–143, 2000.

Mehrdad Ghadiri, Samira Samadi, and Santosh Vempala. Socially fair k-means clustering. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pages 438–448, 2021.

MohammadTaghi Hajiaghayi, Rohit Khandekar, and Guy Kortsarz. Budgeted red-blue median and its generalizations. In *Proceedings of the European Symposium on Algorithms*, pages 314–325, 2010.

Sariel Har-Peled and Sepideh Mahabadi. Near neighbor: Who is the fairest of them all? In *Advances in Neural Information Processing Systems*, pages 13176–13187, 2019.

Moritz Hardt, Eric Price, and Nati Srebro. Equality of opportunity in supervised learning. In *Advances in Neural Information Processing Systems*, pages 3315–3323, 2016.

Lingxiao Huang, Shaofeng Jiang, and Nisheeth Vishnoi. Coresets for clustering with fairness constraints. In *Proceedings of the Conference on Neural Information Processing Systems*, 2019.

Matthew Jones, Huy Nguyen, and Thy Nguyen. Fair k-centers via maximum matching. In *Proceedings of the International Conference on Machine Learning*, pages 4940–4949, 2020.

Christopher Jung, Sampath Kannan, and Neil Lutz. A center in your neighborhood: Fairness in facility location. In *Proceedings of the Symposium on Foundations of Responsible Computing*, page 5:1–5:15, 2020.

Michael Kearns and Aaron Roth. *The ethical algorithm: The science of socially aware algorithm design*. Oxford University Press, 2019.

Jon Kleinberg, Sendhil Mullainathan, and Manish Raghavan. Inherent trade-offs in the fair determination of risk scores. In *Proceedings of the Innovations in Theoretical Computer Science*, 2017.

Matthäus Kleindessner, Pranjal Awasthi, and Jamie Morgenstern. Fair k-center clustering for data summarization. In *Proceedings of the International Conference on Machine Learning*, pages 3448–3457, 2019.

Matthäus Kleindessner, Pranjal Awasthi, and Jamie Morgenstern. A notion of individual fairness for clustering. *arXiv preprint arXiv:2006.04960*, 2020.

Ravishankar Krishnaswamy, Amit Kumar, Viswanath Nagarajan, Yogish Sabharwal, and Barna Saha. The matroid median problem. In *Proceedings of the Symposium on Discrete Algorithms*, pages 1117–1130, 2011.

Ravishankar Krishnaswamy, Shi Li, and Sai Sandeep. Constant approximation for k-median and k-means with outliers via iterative rounding. In *Proceedings of the Symposium on Theory of Computing*, pages 646–659, 2018.

Sepideh Mahabadi and Ali Vakilian. Individual fairness for k-clustering. In *Proceedings of the International Conference on Machine Learning*, pages 6586–6596, 2020.

Konstantin Makarychev, Yury Makarychev, and Ilya Razenshteyn. Performance of Johnson-Lindenstrauss transform for k-means and k-medians clustering. In *Proceedings of the Symposium on Theory of Computing*, pages 1027–1038, 2019.

Evi Micha and Nisarg Shah. Proportionally fair clustering revisited. In *47th International Colloquium on Automata, Languages, and Programming*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik, 2020.

Melanie Schmidt, Chris Schwiegelshohn, and Christian Sohler. Fair coresets and streaming algorithms for fair k-means. In *Proceedings of the International Workshop on Approximation and Online Algorithms*, pages 232–251, 2019.

Roman Vershynin. *High-Dimensional Probability: An Introduction with Applications in Data Science*. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press, 2018. ISBN 9781108244541.

Appendix A. Proofs from Section 4

A.1. Proofs form Section 4.3

Proof of Claim 11 Consider an arbitrary group $j \in [\ell]$.

$$\sum_{u \in P} w_j(u) \cdot \mathcal{R}(u)^p = \sum_{u,v \in P} w_j(u) \cdot d(u,v)^p \cdot x_{uv} \leq z^*$$

where the last inequality holds, since (x, y) is an optimal LP solution for $ClusterLP(\{w_j\}, z_{\boldsymbol{g}}, 2)$ and $ClusterLP(\{w_j\}, z_{\boldsymbol{g}}, 2)$ is a relaxation for fair ℓ_p clustering. ∎

Proof of Claim 12 The proof follows from the consolidation rule of the algorithm. Suppose that u comes before v in the ordering considered by Algorithm 1. Then, since both u, v have non-zero demands and the algorithm has not moved the demand of v to u at the time it processed u, $d(u,v) > \frac{2}{\gamma^{1/p}} \mathcal{R}(v) = \frac{2}{\gamma^{1/p}} \max(\mathcal{R}(u), \mathcal{R}(v))$. ∎

Proof of Lemma 13 In LP relaxations $ClusterLP(\{w_j\}, z, \lambda)$ and $ClusterLP(\{w'_j\}, z, \lambda)$, all constraints other than (5) do not depend on demands and thus are the same in both relaxations. Observe that if $w(v) = 0$ then also $w'(v) = 0$. Therefore, if constraint (5) is present in linear program $ClusterLP(\{w'_j\}, z, \lambda)$ for some u and v, then it is also present in $ClusterLP(\{w_j\}, z, \lambda)$ for the same u and v. We conclude that the set of constraints of $ClusterLP(\{w'_j\}, z, \lambda)$ is a subset of those of $ClusterLP(\{w_j\}, z, \lambda)$. Thus, since (x, y) is a feasible solution for $ClusterLP(\{w_j\}, z, \lambda)$, it is also a feasible solution for $ClusterLP(\{w'_j\}, z, \lambda)$.

Now we show that the LP cost of every group j does not increase. The LP cost of group j is $\sum_{u \in P} w'_j(u)\mathcal{R}(u)$. When Algorithm 1 initializes w'_j in the very beginning (see line 3 of the algorithm), we have $w'_j(u) = w_j(u)$ for all u, and thus at that point $\sum_{u \in P} w'_j(u)\mathcal{R}(u) = \sum_{u \in P} w_j(u)\mathcal{R}(u) \leq z'$. When the algorithm moves demand from u to v on lines 8-9, we always have $R(v) \leq R(u)$. Thus, every time lines 8-9 are executed, the value of expression $\sum_{u \in P} w'_j(u)\mathcal{R}(u)$ may only go down. Therefore, when the cost of LP solution (x, y) with respect to demands w'_j returned by the algorithm is at most z'. ∎

A.2. Proofs from Section 4.3

Proof of Claim 14 Since (i) v' is a closest point in P' to v and (ii) $u \in P'$, we have $d(v, v') \leq d(v, u)$. Applying the triangle inequality, we get

$$d(u, v') \leq d(u, v) + d(v, v') \leq 2d(u, v).$$

∎

Claim 20 *For each $u \in P'$, $\sum_{v \in \mathcal{B}(u, r_u)} x_{uv} \geq 1 - \gamma$ where $r_u = \frac{\mathcal{R}(u)}{\gamma^{1/p}}$.*

Proof

$$\sum_{v \notin \mathcal{B}(u, r_u)} x_{uv} \leq \sum_{v \notin \mathcal{B}(u, r_v)} x_{uv} \frac{d(u, v)^p}{r_u^p} \leq \frac{1}{r_u^p} \sum_{v \in P} x_{uv} d(u, v)^p = \frac{\mathcal{R}_u^p}{r_u^p} = \gamma.$$

Thus, $\sum_{v \in \mathcal{B}(u, r_u)} x_{uv} \geq \sum_{v \in P} x_{uv} - \gamma = 1 - \gamma$. ∎

Proof of Lemma 15 Consider solutions (x, y) and (x', y'). Let $u \in P'$. From Claim 20 and the inequality $x_{uv} \leq y_v$ (for all $v \in P$), we get that

$$\sum_{v \in \mathcal{B}(u, r_u)} y_v \geq 1 - \gamma \qquad \text{where } r_u = \frac{\mathcal{R}(u)}{\gamma^{1/p}}.$$

We now show that for each $v \in \mathcal{B}(u, r_u)$, point u is the closest neighbor of v in P' and, therefore, Algorithm 2 reassigns y_v to point u. Let $\tilde{u}$ be a point in P' other than u. Then,

$$
\begin{aligned}
d(\tilde{u}, v) &\geq d(u, \tilde{u}) - d(u, v) && \triangleright \text{ by the triangle inequality} \\
&> \frac{2}{\gamma^{1/p}} \cdot \mathcal{R}(u) - d(u, v) && \triangleright \text{ from Claim 12, since } u, \tilde{u} \in P' \\
&\geq 2r_u - r_u = r_u \geq d(u, v) && \triangleright \text{ since } v \in \mathcal{B}(u, r_u)
\end{aligned}
$$

17

We conclude that Algorithm 2 assigns at least $\sum_{\mathcal{B}(u,r_u)} y_v \geq 1 - \gamma$ to y'_u. We have, $y'_u \geq 1 - \gamma$ for all $u \in P'$ and $y'_u = 0$ for $u \notin P'$. Therefore, (x', y') is a $(1 - \gamma)$-restricted solution.

Now we upper bound the cost of (x', y'). By Claim 14, when Algorithm 2 moves a center from point $v \notin P'$ to $v' \in P'$, the connection cost to v increases by at most a factor of 2^p: $d(u, v')^p \leq 2^p d(u, v)$ for $u \in P'$. Thus, the cost of (x', y') is at most $2^p z'$.

Lastly, we show that (x', y') is a feasible solution of $ClusterLP(\{w'_j\}, z, 2\lambda)$. Since in each iteration, the value of $\sum_{v \in P} y'_v$ does not increase and the value of $\sum_{v \in P} x'_{uv}$ (for all $u \in P$) does not change, solution (x', y') satisfies constraints (2) and (3). Also, every time we close a center $v \notin P'$ (see lines 4-6 of Algorithm 2), we increase $x'_{uv'}$ by $x_{uv} \leq y_v$ and increase $y'_{v'}$ by y_v (unless doing so would make $y_v > 1$; in this case, we let $y'_v = 1$). Therefore, constraint (4) remains satisfied throughout the execution of the algorithm.

It now remains to show that (x', y') satisfies constraint (5) of $ClusterLP(\{w'_j\}, z, 2\lambda)$. Consider a pair $u, v' \in P'$ with $x'_{uv'} > 0$. First, assume that $x_{uv'} > 0$. Note that (x, y) is a feasible solution of $ClusterLP(\{w'_j\}, z, \lambda)$ and thus satisfies constraint (5). Therefore, $d(u, v') \leq \lambda \Delta_z(u)$, and we are done. Now assume that $x_{uv'} = 0$. This means that we have closed a fractional center $v \in P \setminus P'$ for u and reassigned u from center v to center v'. Then $x_{uv} > 0$ and v' is the closest to v point in P'. Hence,

$$d(u, v') \leq 2d(u, v) \leq 2\lambda \Delta_z(u)$$

where the first inequality holds by Claim 14, and the second inequality holds since (x, y) is a feasible solution for $ClusterLP(\{w'_j\}, z, \lambda)$. We conclude that solution (x', y') is a feasible solution for $ClusterLP(\{w'_j\}, z, 2\lambda)$. ∎

Appendix B. Finding the value of z_g

In this section, we explain why we may assume that we know $z_g \in [z^*, 2z^*]$.

Observation 21 *There exist $u, v \in P$ and $j \in [\ell]$ such that*

$$w_j(u) \cdot d(u, v)^p \leq z^* \leq n \cdot w_j(u) \cdot d(u, v)^p$$

Proof Let C^* be an optimal solution. Assume that group j has the largest cost: $\text{cost}(C^*, w_j) \geq \text{cost}(C^*, w_i)$ for all $i \in [\ell]$. Then, $z^* = \text{cost}(C^*, w_j)$. Consider $u \in P_j$ that maximizes $w_j(u) \cdot d(u, C^*)^p$. Let $v \in C^*$ be the closest center to u. Note that $w_j(u) \cdot d(u, v)^p = w_j(u) \cdot d(u, C^*)^p$ and

$$w_j(u) \cdot d(u, v)^p \leq \text{cost}(C^*, w_j) \leq |P_j| \cdot w_j(u) \cdot d(u, v)^p \leq n \cdot w_j(u) \cdot d(u, v)^p.$$

∎

Now we assume that we have an algorithm that with high probability, returns a β-approximate solution for socially fair ℓ_p-clustering when $z^* \leq z_g \leq 2z^*$. We run this β-approximation algorithm with different estimates for z_g of the form $2^i w_j(u) d(u, v)^p$ where $i \in \{0, \cdots, \lfloor \log_2 n \rfloor\}$, $u, v \in P$, and $j \in [\ell]$. We output the best solution we find. Note that our algorithm runs in polynomial time, since it invokes the β-approximation algorithm at most $O(n^2 \ell \log n)$ times. We note this step can often be significantly sped up; in particular, if the cost is represented as a floating-point number with D binary digits, we can run the β-approximation algorithm at most D times.

Appendix C. Integrality Gap for Linear Relaxation $ClusterLP(\{w_j\}, z, 2)$

In this section, we define what the integrality gap for $ClusterLP(\{w_j\}, z, 2)$ is (this is not necessarily straightforward, since the LP depends on z) and show that the gap is $\Omega(\log \ell / \log \log \ell)$.

Main approximation result restated. The main result of our paper can be formulated as follows (which is implicit in the proof of Theorem 2). Let $f(z)$ be the LP cost of the optimal solution for $ClusterLP(\{w_j\}, z, 2)$ and $g(z) = \max(z, f(z))$. Then the following items hold.

1. For every z, there exists a combinatorial solution of cost at most $e^{O(p)} \frac{\log \ell}{\log \log \ell} g(z)$. Further, this solution can be found in polynomial time.

2. For $z \geq z^*$, $g(z) = z$.

To obtain our result, we take $z_{\boldsymbol{g}} \in [z^*, 2z^*]$. By item 2, $g(z_{\boldsymbol{g}}) = z_g \leq 2z^*$. By item 1, we can find a solution of cost $e^{O(p)} \frac{\log \ell}{\log \log \ell} g(z_{\boldsymbol{g}}) \leq e^{O(p)} \frac{\log \ell}{\log \log \ell} z^*$.

Integrality gap. We now show that dependence on ℓ in item 1 cannot be improved. Namely, we prove that there exists a sequence of instances and parameters z such that every combinatorial solution has cost at least $\Omega \left(\frac{\log \ell}{\log \log \ell} g(z) \right)$.

We construct the following instance for every $k \geq 1$. Let $t = \lfloor \sqrt{k} \rfloor$ and $n = k + t$. Consider a metric space P on n points, in which the distance between every two distinct points is 1. For every set A of t points, create a group $P_A = A$. Note that the total number of groups is $\ell = \binom{n}{t} = e^{\Theta(\sqrt{k} \log k)}$.

Let $z = 1$. Then $\Delta_z(u) = 1$ for every u. Note that constraint 5 is trivially satisfied by any LP solution, since there are simply no two points u and v with $d(u, v) > 2\Delta_z(v) = 2$. Observe that the cost of every solution C of size k is t, since the cost of C for group $P_{P \setminus C} = P \setminus C$ is

$$\sum_{u \in P \setminus C} d(u, C)^p = \sum_{u \in P \setminus C} 1 = t$$

and the cost of every group P_A is at most $|P_A| = t$. Now we construct an LP solution of cost at most 1. For every u, we let $y_u = k/n$, $x_{uu} = k/n$, $x_{uv_u} = 1 - k/n = t/n$ for an arbitrary $v_u \neq u$, and $x_{uv'} = 0$ for $v' \notin \{u, v_u\}$. It is immediate that this is a feasible LP solution. Its cost is the maximum over all groups P_A of

$$\sum_{u \in P_A} \left(\frac{k}{n} \cdot d(u, u)^p + \frac{t}{n} \cdot d(u, v_u)^p \right) = \sum_{u \in P_A} \frac{t}{n} = t^2/n \leq 1.$$

We conclude that $f(z) \leq 1$ and $g(z) = \max(z, f(z)) = 1$. Therefore, the integrality gap of this instance is

$$t/1 = \lfloor \sqrt{k} \rfloor = \Theta \left(\frac{\log \ell}{\log \log \ell} \right).$$