Generalizing Fair Clustering to Multiple Groups: Algorithms and Applications

Diptarka Chakraborty[1], **Kushagra Chatterjee**[1], **Debarati Das**[2], **Tien-Long Nguyen**[2]

[1] Nationaly University of Singapore
[2] Pennsylvania State University
diptarka@comp.nus.edu.sg, kushagra.chatterjee@u.nus.edu, {dxd5606,tfn5179}@psu.edu

Abstract

Clustering is a fundamental task in machine learning and data analysis, but it frequently fails to provide fair representation for various marginalized communities defined by multiple protected attributes – a shortcoming often caused by biases in the training data. As a result, there is a growing need to enhance the fairness of clustering outcomes, ideally by making minimal modifications, possibly as a post-processing step after conventional clustering. Recently, Chakraborty et al. [COLT'25] initiated the study of *closest fair clustering*, though in a restricted scenario where data points belong to only two groups. In practice, however, data points are typically characterized by many groups, reflecting diverse protected attributes such as age, ethnicity, gender, etc.

In this work, we generalize the study of the *closest fair clustering* problem to settings with an arbitrary number (more than two) of groups. We begin by showing that the problem is NP-hard even when all groups are of equal size – a stark contrast with the two-group case, for which an exact algorithm exists. Next, we propose near-linear time approximation algorithms that efficiently handle arbitrary-sized multiple groups, thereby answering an open question posed by Chakraborty et al. [COLT'25].

Leveraging our closest fair clustering algorithms, we further achieve improved approximation guarantees for the *fair correlation clustering* problem, advancing the state-of-the-art results established by Ahmadian et al. [AISTATS'20] and Ahmadi et al. [2020]. Additionally, we are the first to provide approximation algorithms for the *fair consensus clustering* problem involving multiple (more than two) groups, thus addressing another open direction highlighted by Chakraborty et al. [COLT'25].

Introduction

Clustering, the task of partitioning a set of data points into groups based on their mutual similarity or dissimilarity, stands as a fundamental problem in unsupervised learning and is ubiquitous in applications of machine learning and data analysis. Often, each data point carries certain protected attributes, which can be encoded by assigning a specific color to each point. While traditional clustering algorithms typically succeed in optimizing their target objectives, they frequently fail to ensure *fairness* in their results. This shortfall can introduce or perpetuate biases against marginalized groups defined by sensitive attributes such as gender or race (Kay, Matuszek, and Munson 2015; Bolukbasi et al. 2016). These potential biases arise not necessarily from the algorithms themselves, but from historical marginalization inherent in the data used for training. Addressing and mitigating such biases to achieve fair outcomes has emerged as a central topic in the field, with considerable attention given in recent years to the development of algorithms that guarantee *demographic parity* (Dwork et al. 2012) and/or *equal opportunity* (Hardt, Price, and Srebro 2016).

In the context of clustering, (Chierichetti et al. 2017) initiated the study of *fair clustering* to address the issue of disparate impact and promote fair representation. Their work initially focused on datasets with two groups, each point colored either red or blue, and aimed to partition the data such that the blue to red ratio in every cluster matched that of the overall dataset. However, restricting attention to only two colors is limiting, as real-world data often involves multiple (and sometimes non-binary) protected attributes, such as age, race, or gender, resulting in several disjoint colored groups. Subsequent research generalized the fair clustering framework to accommodate more than two colors (Rösner and Schmidt 2018), requiring that the proportion of each colored group within clusters reflects the global proportions of colors. Further studies have considered scenarios where each color group is of equal size (Böhm et al. 2020). We refer to the related works for different variants of the clustering problems that have been studied under fairness constraints.

As previously highlighted, a range of effective clustering algorithms are available for various clustering paradigms; however, these methods may yield unfair or biased results when the training data itself is biased. Such skewed clustering outcomes may lead to inequitable treatment, particularly if the clusters serve as the basis for decision-making or analysis. To counteract this, post-processing existing clustering solutions to mitigate bias and achieve fairness is often necessary, ideally with only minimal adjustments to the cluster assignments. Despite the fundamental nature of this problem, it has received limited attention within the broader context of clustering in prior research. However, it has been studied for specific metric spaces, such as ranking (Celis, Straszak, and Vishnoi 2018; Chakraborty et al. 2022; Kliachkin et al.

2024). Only recently (Chakraborty et al. 2025a) did introduce the problem of obtaining a *closest fair clustering* where given an existing clustering, finding a fair clustering by altering as few assignments as possible. Their study, however, was confined to the special case of two colored groups, which, as previously noted, is restrictive in many practical scenarios. In their work, (Chakraborty et al. 2025a) presented $O(1)$-approximation algorithms for the case of blue and red groups with arbitrary ratios, and demonstrated that the problem can be solved exactly in near-linear time when the groups are of equal size. In the end, they posed the general case of more than two colored groups as an intriguing open direction. In this paper, we address this open problem by devising approximation algorithms and establishing computational hardness that holds even for equi-proportioned multiple-colored groups, thereby showing a stark distinction from the two-group case.

Building upon our findings for the closest fair clustering problem, we further investigate their implications for other prominent clustering variants, specifically *correlation clustering* and *consensus clustering*. In correlation clustering, one is presented with a labeled (typically complete) graph where each edge is marked as either $+$ or $-$. The cost of a clustering is calculated as the total number of $+$ edges that span across clusters and $-$ edges that fall within clusters. This problem has widespread applications in fields such as data mining, social network analysis, computational biology, and marketing analysis (Bonchi, Garcia-Soriano, and Liberty 2014; Hou et al. 2016; Veldt, Gleich, and Wirth 2018; Bressan et al. 2019; Kushagra, Ben-David, and Ilyas 2019). The *fair correlation clustering* problem, introduced in (Ahmadian et al. 2020; Ahmadi et al. 2020), seeks a fair clustering that minimizes this cost. In their work, approximation algorithms were proposed for cases involving multiple colored groups, with the approximation factor depending on the ratios of the group sizes. In this paper, we improved upon their approximation bound and are the first to achieve an approximation guarantee independent of the group size ratio.

In consensus clustering, the objective is to derive a single, representative (consensus) clustering from a collection of clusterings over the same set of data points, to minimize a chosen objective function. The objective often depends on the application, with the most common being the *median* that minimizes the total distance to all input clusterings and the *center* that minimizes the maximum distance. Here, the distance between two clusterings is typically defined by the number of point pairs that are co-clustered (together) in one clustering but not in the other. Consensus clustering is widely applicable across various fields, including bioinformatics (Filkov and Skiena 2004b,a), data mining (Topchy, Jain, and Punch 2003), and community detection (Lancichinetti and Fortunato 2012). Recently, (Chakraborty et al. 2025a) extended the study of consensus clustering to incorporate fairness constraints, providing constant-factor approximations but only in the special case of two colored groups. In this work, we expand upon these results by establishing approximation guarantees for consensus clustering problems involving more than two colored groups.

Our Contribution

In this work, we present both new algorithms and hardness results for the *Closest Fair Clustering* problem in settings where the input data points come from multiple groups, and also study two well-known applications, namely, the *Fair Correlation Clustering* and *Fair Consensus Clustering* problems, and provide new algorithm results for both. Below, we summarize our main contributions.

Closest Fair Clustering with Multiple Groups. In this problem, we are given a clustering $\mathcal{D} = \{D_1, D_2, \ldots, D_m\}$ defined on a dataset V where V is classified into a set χ of disjoint groups, each represented by a unique color. The objective is to compute a fair clustering of V, where the proportion of data points from each group (or color) within any output cluster should reflect their overall proportions in the entire dataset, while also minimizing the distance to the input clustering $\mathcal{D}$.

- Considering $|\chi|$ to be the total number of colors, our first algorithm handles the case where each output cluster must contain the different colored points according to global ratio $p_1 : p_2 : \cdots : p_{|\chi|}$, and achieves an $O(|\chi|^{3.81})$-approximation.

  This study significantly generalizes the work of (Chakraborty et al. 2025a) [COLT '25], which focused solely on the binary (two-group) setting.
- Next, we consider the special case where each output cluster must contain an equal number of data points from each color group, and we provide an $O(|\chi|^{1.6}\log^{2.81}|\chi|)$-approximation for the Closest Fair Clustering problem. Furthermore, when $|\chi|$ is a power of two, we present an improved algorithm with an $O(|\chi|^{1.6})$-approximation.
- Finally, we show that the problem is NP-hard for any setting involving more than two colors, even when all color groups are equally represented in an output cluster.

  This shows a clear hardness gap between the two-color setting, where an exact algorithm exists, and the multi-color case, where the problem becomes computationally intractable; thus underscoring the necessity of our approximation algorithms.

Application to Fair Correlation Clustering. Building on the above result, we study their implications for another key clustering variant: the *Fair Correlation Clustering* problem. In correlation clustering, the input is a graph with edges labeled as + or -, and the goal is to produce a clustering minimizing disagreements; + edges between clusters and - edges within clusters. In the fair version, the clustering must also satisfy group fairness constraints. This problem has been shown to be NP-hard (Ahmadi et al. 2020).

- We begin by designing an algorithm for the setting where each output cluster must contain points from different color groups as per the global ratio $p_1 : p_2 : \cdots : p_{|\chi|}$, achieving an $O(|\chi|^{3.81})$-approximation.

  This eliminates the dependence on the max-min color ratio $q = \frac{\max(p_j)}{\min(p_j)}$ that appeared in the previous $O(q^2|\chi|^2)$

bound of (Ahmadian et al. 2020; Ahmadi et al. 2020), where q can be as large as a polynomial in $|V|$.

- For the special case where each output cluster must contain an equal number of data points from each color group, we improve the approximation to $O(|\chi|^{1.6}\log^{2.81}|\chi|)$, and further to $O(|\chi|^{1.6})$ when $|\chi|$ is a power of two.

  This improves upon the previous $O(|\chi|^2)$ bound given by (Ahmadian et al. 2020; Ahmadi et al. 2020).

Application to Fair Consensus Clustering. Next, we turn to another application: the *consensus clustering* problem. The goal here is to compute a single representative (consensus) clustering from a collection of input clusterings over the same dataset, minimizing a specified objective function (e.g., median or center) while also satisfying group fairness constraints. By combining a triangle inequality argument with our results for Closest Fair Clustering, we obtain new approximation guarantees for this problem. These results generalize the work of (Chakraborty et al. 2025a), which was limited to the binary (two-group) setting, to the more general multi-group case.

- Analogous to our results for fair correlation clustering, we obtain an $O(|\chi|^{3.81})$-approximation algorithm for the general case with arbitrary group proportions.
- For the equi-proportion case, we again improve the approximation to $O(|\chi|^{1.6}\log^{2.81}|\chi|)$, and further to $O(|\chi|^{1.6})$ when $|\chi|$ is a power of two.

The details are provided in the appendix.

Related Works

Since the introduction of the fair clustering in (Chierichetti et al. 2017), recent years have witnessed a significant increase in research focused on different aspects of fair clustering problems. The literature so far encompasses numerous variants of the fair clustering problem with multiple colored groups, such as k-center/median/means clustering (Chierichetti et al. 2017; Huang, Jiang, and Vishnoi 2019), scalable clustering (Backurs et al. 2019), proportional clustering (Chen et al. 2019), fair representational clustering (Bera et al. 2019; Bercea et al. 2019), pairwise fair clustering (Bandyapadhyay, Fomin, and Simonov 2024; Bandyapadhyay et al. 2024, 2025), correlation clustering (Ahmadian et al. 2020; Ahmadi et al. 2020; Ahmadian and Negahbani 2023), 1-clustering over rankings (Wei et al. 2022; Chakraborty et al. 2022, 2025b), and consensus clustering (Chakraborty et al. 2025a), among others.

A comprehensive study of the correlation clustering problem was first undertaken in (Bansal, Blum, and Chawla 2004). Since then, correlation clustering has been studied across various graph settings, including the extensively examined complete graphs (Ailon, Charikar, and Newman 2008; Chawla et al. 2015) and weighted graphs (Demaine et al. 2006). The problem, even when restricted to complete graphs, is known to be APX-hard (Charikar, Guruswami, and Wirth 2005), with the best-known approximation algorithm currently achieving a 1.438-approximation factor (Cao et al. 2024). Its fair variant remains NP-hard even in the case of two color groups of equal sizes (Ahmadi et al. 2020), and several approximation algorithms have been developed for both the exact fairness notion (Ahmadian et al. 2020; Ahmadi et al. 2020) and the relaxed fairness notion (Ahmadian and Negahbani 2023).

The consensus clustering problem, under both the median and center objectives, is known to be NP-hard (Křivánek and Morávek 1986; Swamy 2004) and in fact, APX-hard (that is it is unlikely to have an $(1+\varepsilon)$-factor algorithm for any $\varepsilon > 0$) even with as few as three input clusterings (Bonizzoni et al. 2008). Currently, the best-known algorithms include an $11/7$-approximation for the median objective (Ailon, Charikar, and Newman 2008) and an approximation slightly better than 2 for the center objective (Das and Kumar 2025). Apart from that, various heuristics have been proposed to produce reasonable solutions (e.g.,(Goder and Filkov 2008; Monti et al. 2003; Wu et al. 2014)). More recently,(Chakraborty et al. 2025a) began examining fair consensus clustering, focusing on only two colored groups.

Preliminaries

In this section, we define key terms and concepts that are essential for understanding our proofs and algorithms.

Definition 1 (*Fair Clustering*)**.** Given a set of points V and a set of colors $\chi = \{c_1, c_2, \ldots, c_k\}$, suppose $c_j(V) \subseteq V$ be the set of points of color c_j in V. We call a clustering $\mathcal{F}$ of V a *Fair Clustering* if for all clusters $F \in \mathcal{F}$ we have

$$|c_1(F)| : \cdots : |c_k(F)| = |c_1(V)| : \cdots : |c_k(V)|\,.$$

For two clustering $\mathcal{C}$ and $\mathcal{C}'$ of V we define $\mathrm{dist}(\mathcal{C}, \mathcal{C}')$ as the distance between two clustering $\mathcal{C}$ and $\mathcal{C}'$. The distance is measured by the number of pairs (u, v) that are together in $\mathcal{C}$ but separated by $\mathcal{C}'$ and the number of pairs (u, v) that are separated by $\mathcal{C}$ but together in $\mathcal{C}'$. More specifically,

$$\mathrm{dist}(\mathcal{C}, \mathcal{C}') = \Big|\big\{\{u, v\} \mid u, v \in V,\ [u \sim_{\mathcal{C}} v \wedge u \not\sim_{\mathcal{C}'} v] \vee [u \not\sim_{\mathcal{C}} v \wedge u \sim_{\mathcal{C}'} v]\big\}\Big|$$

where $u \sim_{\mathcal{C}} v$ denotes whether both u and v belong to the same cluster in $\mathcal{C}$ or not.

Definition 2 (Closest *Fair Clustering*)**.** Given an arbitrary clustering $\mathcal{D}$, a clustering $\mathcal{F}^*_{\mathcal{D}}$ is called a closest *Fair Clustering* to $\mathcal{D}$ if for all *Fair Clustering* $\mathcal{F}$ we have $\mathrm{dist}(\mathcal{D}, \mathcal{F}) \geq \mathrm{dist}(\mathcal{D}, \mathcal{F}^*_{\mathcal{D}})$.

We denote a closest *Fair Clustering* to $\mathcal{D}$ by the notation $\mathcal{F}^*_{\mathcal{D}}$.

γ-close *Fair Clustering* We call a *Fair Clustering* $\mathcal{F}$ a γ-close *Fair Clustering* to a clustering $\mathcal{D}$ if

$$\mathrm{dist}(\mathcal{D}, \mathcal{F}) \leq \gamma\,\mathrm{dist}(\mathcal{D}, \mathcal{F}^*_{\mathcal{D}}).$$

Approximate Closest Fair Clustering for Equi-Proportion Groups

In this section, we provide an approximation algorithm to find a closest *Fair Clustering* when all the groups are of equal size.

Theorem 1. *There exists an algorithm that, given a clustering $\mathcal{D}$ where each color group contains an equal number of points, computes a $O(|\chi|^{1.6}\log^{2.81}|\chi|)$-close Fair Clustering in $O(|V|\log|V|)$ time, where χ denotes the set of colors. Moreover, when $|\chi|$ is a power of two, the algorithm computes a $O(|\chi|^{1.6})$-close Fair Clustering.*

First, let us handle the case when $|\chi|$ is a power of 2. To do that, we provide an algorithm `fairpower-of-two`, which produces an $O(|\chi|^{1.6})$-close fair clustering $\mathcal{F}_{\text{fpt}}$ to a clustering $\mathcal{D}$ when $|\chi|$ is a power of 2.

Overview of the Algorithm `fairpower-of-two`: Let the input be a clustering $\mathcal{D} = \{D_1, D_2, \ldots, D_m\}$, where each point is colored from a color set $\chi = \{c_1, \ldots, c_k\}$, and assume $|\chi|$ is a power of two. The goal is to output a clustering $\mathcal{F}_{\text{fpt}}$ in which every cluster contains an equal number of points of each color, i.e., $c_p(F_a) = c_q(F_a)$ for all $p \neq q$, $F_a \in \mathcal{F}_{\text{fpt}}$.

The algorithm proceeds in $\log|\chi|$ iterations. At iteration i, the color set χ is partitioned into $|\chi|/2^i$ disjoint blocks of size 2^i, defined as:

$$B_j^i = \{c_{(j-1)\cdot 2^i+1}, \ldots, c_{j\cdot 2^i}\}, \quad \text{for } j = 1, \ldots, |\chi|/2^i.$$

Let $\mathcal{N}^i$ be the clustering at iteration i, with $\mathcal{N}^0 := \mathcal{D}$. The algorithm maintains the invariant that, in every cluster $N_a^i \in \mathcal{N}^i$, the colors within each block B_j^i appear equally.

Surplus Definition: For adjacent blocks B_j^i and B_{j+1}^i in a cluster N_a^i, the surplus T_a^j is the excess of the larger bucket over the smaller. The surplus is chosen so that all colors in the surplus are equally represented.

Algorithm `fairpower-of-two`:

1. **Initialization:** Set $\mathcal{N}^0 \leftarrow \mathcal{D}$.
2. **Iterative Refinement:** For each iteration $i = 1$ to $\log|\chi|$:
   - Initialize $\mathcal{N}^i \leftarrow \mathcal{N}^{i-1}$.
   - For each pair of disjoint adjacent blocks (B_j^i, B_{j+1}^i):
     - For each cluster $N_a^i \in \mathcal{N}^i$ and odd j, compute the surplus T_a^j between adjacent blocks $B_j^i(N_a^i)$ and $B_{j+1}^i(N_a^i)$, and remove it. Here for a color block B_j^i, $B_j^i(N_a^i)$ is the set of points in N_a^i that has a color from the block B_j^i.
     - Store removed surpluses into sets S_j or S_{j+1}, depending on which block had the surplus.
     - Call `multi-GM`(S_j, S_{j+1}) to form new fair clusters and add them to $\mathcal{N}^i$.
3. **Output:** Return $\mathcal{N}^{\log|\chi|} \rightarrow \mathcal{F}_{\text{fpt}}$.

Subroutine `multi-GM`: Given two collections of point groups from blocks B_j^i and B_{j+1}^i, the `multi-GM` procedure greedily merges pairs of subsets into fair subsets in which each color from $B_j^i \cup B_{j+1}^i$ is equally represented.

The subroutine:

- Iteratively selects one subset from each collection.
- Trims the larger subset to match the smaller, preserving equal color counts.
- Merges the trimmed subsets into a fair set and adds it to the output.

Continue this until no further fair subsets can be formed.

We provide the pseudocode of the algorithms `fairpower-of-two` and `multi-gm` in the appendix.

Proof of Theorem 1

In this section, we analyze the algorithm `fairpower-of-two` by first establishing Lemma 1 stated below,

Lemma 1. *Given a clustering $\mathcal{D}$ as input, the algorithm `fairpower-of-two` computes a $O(|\chi|^{1.6})$-close Fair Clustering, where $|\chi|$ is a power of 2.*

We show the above result by generating a sequence of $\log|\chi|$ intermediate clusterings, with each intermediate step involving an approximation factor of 2, and thus finally achieving a factor of $3^{\log|\chi|} = |\chi|^{1.6}$. We provide the proof in the appendix.

The above Lemma 1 proves Theorem 1 when $|\chi|$ is a power of 2. Now, we prove Theorem 1 for any values of $|\chi|$. To do that, we need to describe the algorithm `make-pdc-fair`.

Overview of the Algorithm `make-pdc-fair`: Given a clustering $\mathcal{I} = \{I_1, I_2, \ldots, I_s\}$ over a point set V, where each point is colored from $\zeta = \{z_1, \ldots, z_r\}$ and satisfies the global proportion:

$$z_1(V) : z_2(V) : \cdots : z_r(V) = p_1 : p_2 : \cdots : p_r,$$

the goal is to construct a fair clustering $\mathcal{F}_{\text{mpf}}$ such that every cluster $F \in \mathcal{F}_{\text{mpf}}$ satisfies this ratio. We assume w.l.o.g. that $p_1 > p_2 > \cdots > p_r$. Also assume that each input cluster is *p-divisible*, i.e., $z_j(I_i)$ is a multiple of p_j. We call such clustering as *p-divisble clustering*(`pdc` for short).

Algorithm `make-pdc-fair`:

1. The algorithm proceeds in $T = \lceil \log_2 r \rceil$ iterations. Let $\mathcal{F}^0 := \mathcal{I}$, and define:

   $$\mathcal{I} = \mathcal{F}^0 \rightarrow \mathcal{F}^1 \rightarrow \cdots \rightarrow \mathcal{F}^T = \mathcal{F}_{\text{mpf}},$$

   where each $\mathcal{F}^t$ enforces proportionality within blocks of colors.
2. **Hierarchical Block Structure:** At iteration t, the color set is partitioned into blocks $\{B_1^t, B_2^t, \ldots, B_{m_t}^t\}$, constructed hierarchically:

   $$B_i^t = B_{2i-1}^{t-1} \cup B_{2i}^{t-1}, \quad \text{with singleton blocks } B_j^0 = \{z_j\}.$$

   If m_{t-1} is odd, the last block is carried forward unchanged.
3. **Balancing Rule:** To merge two sub-blocks $A = B_{2i-1}^{t-1}$ and $B = B_{2i}^{t-1}$, consider a cluster $F^{t-1} \in \mathcal{F}^{t-1}$, where

   $$z_c(F^{t-1}) = p_c \cdot x \text{ for } z_c \in A, \quad z_d(F^{t-1}) = p_d \cdot y \text{ for } z_d \in B.$$

   To equalize the scaling factors x and y, we do:
   - If $x > y$: merge $p_d \cdot (x - y)$ points of color $z_d \in B$ into F^{t-1}.

- If $x < y$: cut $p_d \cdot (y - x)$ points of color $z_d \in B$ from F^{t-1}.

This ensures that the merged block $B_i^t = A \cup B$ in each cluster $F^t \in \mathcal{F}^t$ satisfies the combined proportionality.

4. **Output:** After $T = \lceil \log_2 r \rceil$ iterations, the final clustering $\mathcal{F}_{\text{mpf}}$ satisfies: For all $F \in \mathcal{F}_{\text{mpf}}$,

$$z_1(F) : z_2(F) : \cdots : z_r(F) = p_1 : p_2 : \cdots : p_r.$$

We provide the pseudocode of `make-pdc-fair` in the appendix.

To analyse the algorithm `make-pdc-fair` we need to prove the following lemma.

Lemma 2. *The algorithm `make-pdc-fair` outputs a clustering $\mathcal{F}$ that is $O(r^{2.81})$-close Fair Clustering to the input clustering $\mathcal{I}$, where r is the number of colors.*

We show the above result by generating a sequence of $\log r$ intermediate clusterings, with each intermediate step involving an approximation factor of 6, and thus finally achieving a factor of $7^{\log r} = r^{2.81}$. We provide a detailed proof in the appendix.

Using Lemma 1 and Lemma 2, we can prove Theorem 1. We provide the full proof of Theorem 1 in the appendix.

Proof Sketch of Theorem 1. We design an algorithm `fair-equi` to convert an arbitrary clustering $\mathcal{D}$, where the color set χ is not necessarily a power of two, into a fair clustering $\mathcal{F}$ in which every color appears equally in each cluster. The algorithm proceeds in two main stages:

1. **Color Grouping and Intermediate Fairness:**
   - Partition the color set χ into $\log |\chi|$ disjoint groups $G_1, \ldots, G_{\log |\chi|}$, where each group's size is a power of two. This is done greedily by assigning group sizes according to the binary representation of $|\chi|$.
   - Apply the algorithm `fairpower-of-two` to obtain an intermediate clustering $\mathcal{I}$, in which each group G_ℓ satisfies intra-group fairness: all colors in G_ℓ appear equally in each cluster.
2. **Global Fairness via Multi-Group Merging:**
   - Treat each group G_ℓ as a single meta-color and apply the algorithm `make-pdc-fair` on $\mathcal{I}$ to obtain the final clustering $\mathcal{F}$.
   - `make-pdc-fair` ensures that across all clusters, the meta-colors (i.e., groups) are in proportion to their sizes, and also restores uniformity within each group, thus achieving full color-wise fairness.

Approximation Bound:

- By Lemma 1, the intermediate clustering $\mathcal{I}$ is $O(|\chi|^{1.6})$-close to $\mathcal{D}$.
- By Lemma 2, the final clustering $\mathcal{F}$ is $O(\log^{2.81} |\chi|)$-close to $\mathcal{I}$.
- Combining via triangle inequality yields:

$$\text{dist}(\mathcal{D}, \mathcal{F}) \leq O(|\chi|^{1.6} \log^{2.81} |\chi|) \cdot \text{dist}(\mathcal{D}, \mathcal{F}^*_{\mathcal{D}})$$

where $\mathcal{F}^*_{\mathcal{D}}$ is the closest fair clustering to $\mathcal{D}$.

This completes the proof sketch. □

Approximate Closest Fair Clustering for Arbitrary-Proportion Groups

In this section, we prove the following theorem.

Theorem 2. *There is an algorithm that, given an arbitrary clustering $\mathcal{D}$ over a vertex set V where each vertex $v \in V$ has a color in $\chi = \{c_1, \ldots, c_k\}$, finds a $O(|\chi|^{3.81})$-close Fair Clustering $\mathcal{F}$ in time $O(|V| \log |V|)$.*

To prove the above theorem, we take a 2-step approach similar to (Chakraborty et al. 2025a), which was constrained to only two colors.

(i) First, we convert the clustering $\mathcal{D}$ to a clustering $\mathcal{M}$ such that for each cluster $M_i \in \mathcal{M}$, $c_j(M_i)$ is divisible by p_j. Recall we call such a clustering a p-divisible clustering.

(ii) In the second step, we will provide the clustering $\mathcal{M}$ as input to the algorithm `make-pdc-fair` and get a fair clustering $\mathcal{F}$ as output.

Now, we provide an algorithm `create-pdc` to convert a clustering $\mathcal{D}$ on a vertex set V to a p-divisible clustering $\mathcal{M}$.

Overview of the algorithm `create-pdc`: Input: A clustering $\mathcal{D} = \{D_1, \ldots, D_m\}$ over vertex set V, color set $\chi = \{c_1, \ldots, c_k\}$, and a proportion vector $\mathbf{p} = (p_1, \ldots, p_k)$ satisfying:

$$c_1(V) : c_2(V) : \cdots : c_k(V) = p_1 : p_2 : \cdots : p_k.$$

Goal: Convert $\mathcal{D}$ into a p-divisible clustering $\mathcal{M}$ where each cluster contains a multiple of p_j points of color c_j.

Key Definitions:

- Surplus: $\sigma(D_i, c_j) \subseteq D_i$ of size $c_j(D_i) \bmod p_j$ if $p_j \nmid c_j(D_i)$, else it has size p_j, denoting excess c_j-colored vertices in D_i.
- Total surplus: $\sigma_j = \sum_{D_i \in \mathcal{D}} |\sigma(D_i, c_j)|$ (always a multiple of p_j).
- Deficit: $\delta(D_i, c_j) \subseteq V \setminus D_i$, of size $p_j - |\sigma(D_i, c_j)|$. Represents the number of c_j colored points required to make it a multiple of p_j.
- Cut and merge costs:

$$\kappa^j(D_i) = |\sigma(D_i, c_j)| \cdot (|D_i| - |\sigma(D_i, c_j)|)$$
$$\mu^j(D_i) = |\delta(D_i, c_j)| \cdot |D_i|$$

Algorithm `create-pdc`:

1. For each color c_j, initialize σ_j / p_j empty auxiliary clusters $\{P_1, \ldots, P_{\sigma_j/p_j}\}$.
2. Classify each cluster $D_i \in \mathcal{D}$ into:
   - `CUT`, if $|\sigma(D_i, c_j)| \leq p_j/2$
   - `MERGE`, otherwise.
3. **Cut and redistribute:** While `CUT` is non-empty:
   - For $D_i \in$ `CUT`, remove $\sigma(D_i, c_j)$ from D_k.
   - Try to donate surplus to deficits in $D_\ell \in$ `MERGE`.
   - If no deficit remains, assign surplus to an available extra cluster P_m, ensuring each P_m reaches size p_j.

4. **Handle remaining merges:** While deficits remain:
   - Pick the cluster with minimal $\kappa^j(D_k) - \mu^j(D_k)$, redistribute its surplus as above. Note that in this step, we can pick a cluster multiple times.
   - Remove a cluster $D_\ell \in \texttt{MERGE}$ if it deficit is satisfied.
   - Repeat until all deficits are filled.
5. **Output:** The final clustering $\mathcal{M}$ where each cluster is p-divisible for all colors.

We provide the pseudocode of the algorithm `create-pdc` in the appendix.

Now, we analyze the algorithm `create-pdc`. To analyze and state the lemma, let us define some terms

- **Optimal-p-divisible clustering**: Given a clustering $\mathcal{D}$, we call a clustering $\mathcal{M}^*$ an optimal-p-divisible clustering if $\text{dist}(\mathcal{D}, \mathcal{M}^*) \leq \text{dist}(\mathcal{D}, \mathcal{M})$.
- **α-close-p-divisible clustering**: Given a clustering $\mathcal{D}$, we call a clustering $\mathcal{M}$ an α-close-p-divisible clustering if $\text{dist}(\mathcal{D}, \mathcal{M}) \leq \alpha\, \text{dist}(\mathcal{D}, \mathcal{M}^*)$.

Now, we state the lemma for analysing the algorithm `create-pdc`.

Lemma 3. *Given a clustering $\mathcal{D}$, the algorithm `create-pdc` outputs a clustering $\mathcal{M}$ such that $\mathcal{M}$ is $O(|\chi|)$-close-p-divisible clustering to $\mathcal{D}$.*

We provide the proof of Lemma 3 in the appendix. Next, we prove Theorem 2 assuming Lemma 3.

Proof of Theorem 2. To prove the theorem, we describe an algorithm `fair-general` that proceeds in two stages. First, we apply the algorithm `create-pdc` to the input clustering $\mathcal{D}$ to obtain a p-divisible clustering $\mathcal{M}$ that is $O(|\chi|)$-close to $\mathcal{D}$. Then, we apply the algorithm `make-pdc-fair` on $\mathcal{M}$ to obtain the final fair clustering $\mathcal{F}$, which is $O(|\chi|^{2.81})$-close to $\mathcal{M}$ (see Lemma 2). By combining the guarantees from both steps, we conclude that $\mathcal{F}$ is $O(|\chi|^{3.81})$-close *Fair Clustering* to the input $\mathcal{D}$.

$$
\begin{aligned}
\text{dist}(\mathcal{D}, \mathcal{F}) &\leq \text{dist}(\mathcal{D}, \mathcal{M}) + \text{dist}(\mathcal{M}, \mathcal{F}) \text{ (triangle inequality)} \\
&\leq O(|\chi|)\, \text{dist}(\mathcal{D}, \mathcal{F}^*) + O(|\chi|^{2.81})\, \text{dist}(\mathcal{M}, \mathcal{F}^*) \\
&\leq O(|\chi|)\, \text{dist}(\mathcal{D}, \mathcal{F}^*) + O(|\chi|^{2.81})(\text{dist}(\mathcal{D}, \mathcal{M}) \\
&\quad + \text{dist}(\mathcal{D}, \mathcal{F}^*)) \text{ (triangle inequality)} \\
&\leq O(|\chi| + |\chi|^{3.81} + |\chi|^{2.81})\, \text{dist}(\mathcal{D}, \mathcal{F}^*) \\
&\leq O(|\chi|^{3.81})\, \text{dist}(\mathcal{D}, \mathcal{F}^*)
\end{aligned}
$$

This completes the proof of Theorem 2. □

Hardness for Three Equi-Proportion Groups

In this section, we show that finding a closest fair clustering to a given clustering where each point is assigned one color from a set of $k \geq 3$ colors is hard even when the number of points in each color class is equal. Our reduction also extends to arbitrary color ratios.

We begin by defining the decision version of closest fair clustering with multiple colors and equal representation.

Definition 3 (k-CLOSEST EQUIFAIR)**.** Given a clustering $\mathcal{H}$ over a set of points V where each point is assigned with one of the colors from a set of $k \geq 3$ colors, and the numbers of points of each color are equal, together with a non-negative integer τ, decide between the following:

- YES: There exists a *Fair Clustering* (on input point set) $\mathcal{F}$ such that $\text{dist}(\mathcal{H}, \mathcal{F}) \leq \tau$;
- NO: For every *Fair Clustering* (on input point set) $\mathcal{F}$, $\text{dist}(\mathcal{H}, \mathcal{F}) > \tau$.

We show the following theorem.

Theorem 3. *For any integer $k \geq 3$, k-CLOSEST EQUIFAIR is* **NP***-hard.*

We present a polynomial-time reduction from the 3-PARTITION problem (defined below) to k-CLOSEST EQUIFAIR.

Definition 4 (3-PARTITION)**.** Given a (multi)set of positive integers $S = \{x_1, \ldots, x_d\}$, decide whether (YES:) there exists a partition of S into m disjoint subsets $S_1, S_2, \ldots, S_f \subseteq S$ where $f = d/3$, such that

- For all i, $|S_i| = 3$; and
- For all i, $\sum_{x_j \in S_i} x_j = T$, where $T = \frac{\sum_{x_j \in S} x_j}{n/3}$,

or (NO:) no such partition exists.

We apply our reduction from a more restricted version of 3-PARTITION, in which each $x_i \in S$ satisfies $x_i \in (T/4, T/2)$, where $T = \frac{3}{n}\sum_{x_i \in S} x_i$, and additionally $x_i \leq d^b$, for some non-negative constant b. Note that, this variant remains **NP**-complete, as established by (Garey and Johnson 1975), which shows that 3-PARTITION as defined in Definition 4 is *strongly* **NP***-complete*. Hence, for the rest of this section, we refer to this restricted version simply as 3-PARTITION.

Reduction from 3-PARTITION to k-CLOSEST EQUIFAIR. Given a 3-PARTITION instance $S = \{x_1, x_2, \ldots, x_d\}$ we create a k-CLOSEST EQUIFAIR instance $(\mathcal{H}, \tau)$ as follows:

- $\mathcal{H} = \{\text{GB}_1, \text{GB}_2, \ldots, \text{GB}_{d/3}, \text{R}_1, \text{R}_2, \ldots, \text{R}_d\}$, where for each $i \in \{1, \ldots, d/3\}$, GB_i is a cluster of size $(k-1)T$ with T points of color c_t $(2 \leq t \leq k)$, and for each $j \in \{1, \ldots, n\}$, R_j is a monochromatic c_1 cluster (i.e., containing only points with color c_1) of size x_j (i.e., $|\text{R}_j| = x_j$);
- Set $\tau = \frac{n}{3}(k-1)T^2 + \frac{1}{2}\sum_{i=1}^{n} x_i(T - x_i)$, if $k > 3$, and, $\tau = 2\sum_{i=1}^{n} x_i^2 + 2\sum_{i=1}^{n} x_i(T - x_i)$, if $k = 3$.

Note that each $x_i \leq d^b$, for some non-negative constant b, the size of the instance $(\mathcal{H}, \tau)$ is polynomial in d. Moreover, it is straightforward to see that the reduction runs in polynomial time.

The following lemma argues that the above reduction maps a YES instance of the 3-PARTITION to a YES instance of the k-CLOSEST EQUIFAIR.

Lemma 4. *For any integer $k \geq 3$, if S is a* YES *instance of the* 3-PARTITION*, then $(\mathcal{H}, \tau)$ is also a* YES *instance of the* k-CLOSEST EQUIFAIR.

We defer the proof of Lemma 4 to the appendix.

It remains to demonstrate that our reduction maps a NO instance of the 3-PARTITION to a NO instance of the k-CLOSEST EQUIFAIR.

Lemma 5. *For $k \geq 3$, if S is a* NO *instance of the* 3-PARTITION*, then $(\mathcal{H}, \tau)$ is also a* NO *instance of the* k-CLOSEST EQUIFAIR.

Proof. Assume to the contrary that $(\mathcal{H}, \tau)$ is a YES instance. Then there exists a *Fair Clustering* $\mathcal{F}$ such that $\mathrm{dist}(\mathcal{H}, \mathcal{F}) \leq \tau$.

Without loss of generality, we refer to c_1 as the color red, and refer to c_2 as the color blue.

Let V' be the set of red-blue points obtained from V by recoloring every point with color c_j ($j \geq 3$) to blue. Denote $\mathcal{H}^c$ and $\mathcal{F}^c$ the clusterings obtained from $\mathcal{H}$ and $\mathcal{F}$, respectively, under this recoloring. Then, $\mathrm{dist}(\mathcal{H}^c, \mathcal{F}^c) = \mathrm{dist}(\mathcal{H}, \mathcal{F})$. We analyze the structure of $\mathcal{H}^c$ and $\mathcal{F}^c$.

Observe that $\mathcal{H}^c$ is a clustering over V', in which $\mathcal{H}^c = \{\mathrm{B}_1, \mathrm{B}_2, \ldots, \mathrm{B}_{n/3}, \mathrm{R}_1, \mathrm{R}_2, \ldots, \mathrm{R}_n\}$, where each B_i is a monochromatic blue cluster of size $(k-1)T$, obtained from the original cluster GB_i in $\mathcal{H}$ by recoloring every point to blue. Each R_i is a monochromatic red cluster, which remains unchanged from $\mathcal{H}$.

Now we claim that $\mathcal{F}^c$ is a *Fair Clustering* clustering over V'. Indeed, recall that $\mathcal{F}$ is a *Fair Clustering* over V. Hence, in every cluster $F \in \mathcal{F}$, the numbers of points of each color are equal, that is, $|c_i(F)| = |c_j(F)|$, for all $1 \leq i, j \leq k$. Note that each cluster $F^c \in \mathcal{F}^c$ is obtained from a cluster in $\mathcal{F}$ by recoloring every point with color c_i ($i \geq 3$) to blue. Hence, the ratio between the number of blue points and the number of red points in F^c is $(k-1)$. This implies that $\mathcal{F}^c$ is a *Fair Clustering* over V'.

Applying a result from (Chakraborty et al. 2025a, Lemma 45, Lemma 46) for the set of points V', it follows that $\mathrm{dist}(\mathcal{H}^c, \mathcal{F}^c) > \tau$, which is a contradiction since we have established that $\mathrm{dist}(\mathcal{H}^c, \mathcal{F}^c) = \mathrm{dist}(\mathcal{H}, \mathcal{F}) \leq \tau$. This concludes that $(\mathcal{H}, \tau)$ is a NO instance. □

As our reduction runs in polynomial time, from Lemma 4 and Lemma 5, we conclude that k-CLOSEST EQUIFAIR is **NP**-hard. This completes the proof of Theorem 3. In the appendix, we remark on how to generalize this reduction for arbitrary ratios.

Implication to Fair Correlation Clustering

Correlation Clustering. Given a complete undirected graph $G(V, E)$ where each edge $(u, v) \in E$ is labeled either "+" (similar) or "−" (dissimilar), let E^+ and E^- denote the sets of "+" and "−" edges, respectively. A clustering $\mathcal{C} = \{C_1, C_2, \ldots, C_t\}$ partitions V into disjoint subsets.

Let $\mathrm{EXT}(\mathcal{C})$ denote the set of inter-cluster edges and $\mathrm{INT}(\mathcal{C})$ the set of intra-cluster edges. The cost of clustering $\mathcal{C}$ is defined as:

$$\mathrm{cost}(\mathcal{C}) = |\,\mathrm{EXT}(\mathcal{C}) \cap E^+| + |\,\mathrm{INT}(\mathcal{C}) \cap E^-|.$$

The goal is to find a clustering that minimizes $\mathrm{cost}(\mathcal{C})$. In addition, when we want $\mathcal{C}$ to be also a *Fair Clustering*, the problem is referred to as *fair correlation clustering*.

Theorem 4. *There exists an algorithm that, given a correlation clustering instance G, computes a $O(|\chi|^{1.6}\log^{2.81}|\chi|)$ approximate fair correlation clustering when there are equal number of data points from each color group, and a $O(|\chi|^{3.81})$ approximate fair correlation clustering when the ratio between the number of points from different color groups is arbitrary.*

To prove Theorem 4, we use the following.

Lemma 6. *Let $\mathcal{C}$ be a clustering, and suppose there exists an algorithm $\mathcal{A}$ that computes an γ-close fair clustering with respect to $\mathcal{C}$. Additionally, suppose there exists a β-approximation algorithm $\mathcal{B}$, for the standard correlation clustering problem on a graph G. Then, there exists an algorithm that computes a fair correlation clustering of G with approximation factor $(\gamma + \beta + \gamma\beta)$.*

Proof. Let us first describe the algorithm `fairifyCC`, which computes a fair correlation clustering for a given instance G.

- **Input:** Correlation clustering instance G.
- **Output:** A fair clustering $\mathcal{F}$.
- **Procedure:**
  1. Compute a correlation clustering $\mathcal{D}$ of G using a β-approximation algorithm $\mathcal{B}$.
  2. Apply the closest fair clustering algorithm $\mathcal{A}$ to $\mathcal{D}$ to obtain a fair clustering $\mathcal{F}$ that is γ-close to $\mathcal{D}$.
  3. Return $\mathcal{F}$.

We argue that $\mathcal{F}$ is $(\gamma + \beta + \gamma\beta)$ approximate correlation clustering of G using triangle inequality. We defer the proof to the appendix. □

Proof of Theorem 4. By (Cao et al. 2025), we get that there exists an $O(1)$-approximation algorithm to find a correlation clustering for a correlation clustering instance G. When each color group has the same size, the algorithm `fair-equi` produces an $O(|\chi|^{1.6}\log^{2.81}|\chi|)$-close clustering to any input clustering $\mathcal{D}$. Hence, by Lemma 6 we get that the algorithm `fairifyCC` produces an $O(|\chi|^{1.6}\log^{2.81}|\chi|)$ approximate fair correlation clustering.

In the general case with arbitrary group size ratio $p_1 : p_2 : \cdots : p_{|\chi|}$, the algorithm `fair-general` gives $O(|\chi|^{3.81})$-close clustering to any input clustering $\mathcal{D}$. Hence, again by Lemma 6 we show algorithm `fairifyCC` produces an $O(|\chi|^{3.81})$ approximate fair correlation clustering for this. □

Conclusion

In this paper, we generalize the closest fair clustering problem originally proposed by (Chakraborty et al. 2025a) [COLT '25] to scenarios involving any number of groups, thereby addressing settings with non-binary, multiple protected attributes. We demonstrate that the problem becomes NP-hard even when there are just three equal-sized groups, showing a strong separation with the two equi-proportion

group case where an exact solution exists. We further propose near-linear time approximation algorithms for clustering with multiple (potentially unequal-sized) groups, answering an open problem posed by (Chakraborty et al. 2025a) [COLT '25]. Leveraging these results, we achieve improved approximation guarantees for fair correlation clustering and, for the first time, provide approximation guarantees for fair consensus clustering involving more than two groups. Promising directions for future research include improving the approximation factors further and investigating alternative fairness criteria with similar approximation guarantees.

Acknowledgments

Diptarka Chakraborty was supported in part by an MoE AcRF Tier 1 grant (T1 251RES2303), and a Google South & South-East Asia Research Award. Kushagra Chatterjee was supported by an MoE AcRF Tier 1 grant (T1 251RES2303). Debarati Das was supported in part by NSF grant 2337832.

References

Ahmadi, S.; Galhotra, S.; Saha, B.; and Schwartz, R. 2020. Fair correlation clustering. *arXiv preprint arXiv:2002.03508*.

Ahmadian, S.; Epasto, A.; Kumar, R.; and Mahdian, M. 2020. Fair Correlation Clustering. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 108, 4195–4205.

Ahmadian, S.; and Negahbani, M. 2023. Improved approximation for fair correlation clustering. In *International Conference on Artificial Intelligence and Statistics*, 9499–9516. PMLR.

Ailon, N.; Charikar, M.; and Newman, A. 2008. Aggregating inconsistent information: ranking and clustering. *Journal of the ACM (JACM)*, 55(5): 1–27.

Backurs, A.; Indyk, P.; Onak, K.; Schieber, B.; Vakilian, A.; and Wagner, T. 2019. Scalable Fair Clustering. In *International Conference on Machine Learning (ICML)*, volume 97, 405–413.

Bandyapadhyay, S.; Chen, T.; Friggstad, Z.; and Jamshidian, M. 2025. A Constant-Factor Approximation for Pairwise Fair k-Center Clustering. In *International Conference on Integer Programming and Combinatorial Optimization*, 43–57. Springer.

Bandyapadhyay, S.; Chlamtáč, E.; Friggstad, Z.; Jamshidian, M.; Makarychev, Y.; and Vakilian, A. 2024. A Polynomial-Time Approximation for Pairwise Fair k-Median Clustering. *arXiv preprint arXiv:2405.10378*.

Bandyapadhyay, S.; Fomin, F. V.; and Simonov, K. 2024. On coresets for fair clustering in metric and euclidean spaces and their applications. *Journal of Computer and System Sciences*, 142: 103506.

Bansal, N.; Blum, A.; and Chawla, S. 2004. Correlation clustering. *Machine learning*, 56(1): 89–113.

Bera, S.; Chakrabarty, D.; Flores, N.; and Negahbani, M. 2019. Fair algorithms for clustering. *Advances in Neural Information Processing Systems*, 32.

Bercea, I. O.; Groß, M.; Khuller, S.; Kumar, A.; Rösner, C.; Schmidt, D. R.; and Schmidt, M. 2019. On the Cost of Essentially Fair Clusterings. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques (APPROX/RANDOM 2019)*, 18–1. Schloss Dagstuhl–Leibniz-Zentrum für Informatik.

Böhm, M.; Fazzone, A.; Leonardi, S.; and Schwiegelshohn, C. 2020. Fair clustering with multiple colors. *arXiv preprint arXiv:2002.07892*.

Bolukbasi, T.; Chang, K.-W.; Zou, J. Y.; Saligrama, V.; and Kalai, A. T. 2016. Man is to computer programmer as woman is to homemaker? debiasing word embeddings. *Advances in Neural Information Processing Systems (NeurIPS)*, 29.

Bonchi, F.; Garcia-Soriano, D.; and Liberty, E. 2014. Correlation clustering: from theory to practice. In *KDD*, 1972.

Bonizzoni, P.; Vedova, G. D.; Dondi, R.; and Jiang, T. 2008. On the Approximation of Correlation Clustering and Consensus Clustering. *J. Comput. Syst. Sci.*, 74(5): 671–696.

Bressan, M.; Cesa-Bianchi, N.; Paudice, A.; and Vitale, F. 2019. Correlation clustering with adaptive similarity queries. *Advances in neural information processing systems*, 32.

Cao, N.; Cohen-Addad, V.; Lee, E.; Li, S.; Lolck, D. R.; Newman, A.; Thorup, M.; Vogl, L.; Yan, S.; and Zhang, H. 2025. Solving the Correlation Cluster LP in Sublinear Time. In *Proceedings of the 57th Annual ACM Symposium on Theory of Computing*, 1154–1165.

Cao, N.; Cohen-Addad, V.; Lee, E.; Li, S.; Newman, A.; and Vogl, L. 2024. Understanding the Cluster Linear Program for Correlation Clustering. In *Proceedings of the 56th Annual ACM Symposium on Theory of Computing*, 1605–1616.

Celis, L. E.; Straszak, D.; and Vishnoi, N. K. 2018. Ranking with Fairness Constraints. In *International Colloquium on Automata, Languages, and Programming (ICALP)*, volume 107, 28:1–28:15.

Chakraborty, D.; Chatterjee, K.; Das, D.; Nguyen, T. L.; and Nobahari, R. 2025a. Towards Fair Representation: Clustering and Consensus. *38th Annual Conference on Learning Theory (COLT 2025); full version: arXiv preprint arXiv:2506.08673*.

Chakraborty, D.; Das, H.; Dey, S.; and Yan, A. H. Y. 2025b. Improved Rank Aggregation under Fairness Constraint. *arXiv preprint arXiv:2505.10006*.

Chakraborty, D.; Das, S.; Khan, A.; and Subramanian, A. 2022. Fair rank aggregation. *Advances in Neural Information Processing Systems*, 35: 23965–23978. Full version: arXiv preprint arXiv:2308.10499.

Charikar, M.; Guruswami, V.; and Wirth, A. 2005. Clustering with qualitative information. *Journal of Computer and System Sciences*, 71(3): 360–383.

Chawla, S.; Makarychev, K.; Schramm, T.; and Yaroslavtsev, G. 2015. Near optimal lp rounding algorithm for correlationclustering on complete and complete k-partite graphs. In *Proceedings of the forty-seventh annual ACM symposium on Theory of computing*, 219–228.

Chen, X.; Fain, B.; Lyu, L.; and Munagala, K. 2019. Proportionally Fair Clustering. In *International Conference on Machine Learning (ICML)*, 1032–1041.

Chierichetti, F.; Kumar, R.; Lattanzi, S.; and Vassilvitskii, S. 2017. Fair clustering through fairlets. In *Advances in Neural Information Processing Systems (NeurIPS)*, 5029–5037.

Das, D.; and Kumar, A. 2025. Breaking the Two Approximation Barrier for Various Consensus Clustering Problems. In Azar, Y.; and Panigrahi, D., eds., *Proceedings of the 2025 Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2025, New Orleans, LA, USA, January 12-15, 2025*, 323–372. SIAM.

Demaine, E. D.; Emanuel, D.; Fiat, A.; and Immorlica, N. 2006. Correlation clustering in general weighted graphs. *Theoretical Computer Science*, 361(2-3): 172–187.

Dwork, C.; Hardt, M.; Pitassi, T.; Reingold, O.; and Zemel, R. 2012. Fairness through awareness. In *Innovations in Theoretical Computer Science*, 214–226.

Filkov, V.; and Skiena, S. 2004a. Heterogeneous data integration with the consensus clustering formalism. In *International Workshop on Data Integration in the Life Sciences*, 110–123. Springer.

Filkov, V.; and Skiena, S. 2004b. Integrating microarray data by consensus clustering. *International Journal on Artificial Intelligence Tools*, 13(04): 863–880.

Garey, M. R.; and Johnson, D. S. 1975. Complexity results for multiprocessor scheduling under resource constraints. *SIAM journal on Computing*, 4(4): 397–411.

Goder, A.; and Filkov, V. 2008. Consensus clustering algorithms: Comparison and refinement. In *2008 Proceedings of the Tenth Workshop on Algorithm Engineering and Experiments (ALENEX)*, 109–117. SIAM.

Hardt, M.; Price, E.; and Srebro, N. 2016. Equality of opportunity in supervised learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 3315–3323.

Hou, J. P.; Emad, A.; Puleo, G. J.; Ma, J.; and Milenkovic, O. 2016. A new correlation clustering method for cancer mutation analysis. *Bioinformatics*, 32(24): 3717–3728.

Huang, L.; Jiang, S. H.; and Vishnoi, N. K. 2019. Coresets for Clustering with Fairness Constraints. In *Advances in Neural Information Processing Systems (NeurIPS)*, 7587–7598.

Kay, M.; Matuszek, C.; and Munson, S. A. 2015. Unequal representation and gender stereotypes in image search results for occupations. In *ACM conference on human factors in computing systems*, 3819–3828.

Kliachkin, A.; Psaroudaki, E.; Mareček, J.; and Fotakis, D. 2024. Fairness in Ranking: Robustness through Randomization without the Protected Attribute. In *2024 IEEE 40th International Conference on Data Engineering Workshops (ICDEW)*, 201–208. IEEE.

Křivánek, M.; and Morávek, J. 1986. NP-hard problems in hierarchical-tree clustering. *Acta informatica*, 23: 311–323.

Kushagra, S.; Ben-David, S.; and Ilyas, I. 2019. Semi-supervised clustering for de-duplication. In *The 22nd International Conference on Artificial Intelligence and Statistics*, 1659–1667. PMLR.

Lancichinetti, A.; and Fortunato, S. 2012. Consensus clustering in complex networks. *Scientific reports*, 2(1): 336.

Monti, S.; Tamayo, P.; Mesirov, J.; and Golub, T. 2003. Consensus clustering: a resampling-based method for class discovery and visualization of gene expression microarray data. *Machine learning*, 52: 91–118.

Rösner, C.; and Schmidt, M. 2018. Privacy Preserving Clustering with Constraints. In *45th International Colloquium on Automata, Languages, and Programming (ICALP 2018)*, 96–1. Schloss Dagstuhl–Leibniz-Zentrum für Informatik.

Swamy, C. 2004. Correlation Clustering: maximizing agreements via semidefinite programming. In *SODA*, volume 4, 526–527. Citeseer.

Topchy, A.; Jain, A. K.; and Punch, W. 2003. Combining multiple weak clusterings. In *Third IEEE international conference on data mining*, 331–338. IEEE.

Veldt, N.; Gleich, D. F.; and Wirth, A. 2018. A correlation clustering framework for community detection. In *Proceedings of the 2018 World Wide Web Conference*, 439–448.

Wei, D.; Islam, M. M.; Schieber, B.; and Roy, S. B. 2022. Rank Aggregation with Proportionate Fairness. In *SIGMOD International Conference on Management of Data*, 262–275.

Wu, J.; Liu, H.; Xiong, H.; Cao, J.; and Chen, J. 2014. K-means-based consensus clustering: A unified view. *IEEE transactions on knowledge and data engineering*, 27(1): 155–169.

Algorithm 1: `fairpower-of-two`($\mathcal{D}$)

Input: Clustering $\mathcal{D}$
Output: A fair clustering $\mathcal{N}$

Let, $\mathcal{N}^0 = \{N_1^0, N_2^0, \ldots, N_\ell^0\} = \mathcal{D}$.
for $i \leftarrow 1$ **to** $\log |\chi|$ **do**
  $\mathcal{N}^i = \mathcal{N}^{i-1}$
  $j \leftarrow 1$
  while $j \neq |\chi|/2^i$ **do**
    $S_j, S_{j+1} \leftarrow \varnothing$
    foreach $N_a^i \in \mathcal{N}^i$ **do**
      $N_a^i \leftarrow N_a^i \setminus T_a^j$
      if $|B_j^i(N_a^i)| \geq |B_{j+1}^i(N_a^i)|$ **then**
        $S_j \leftarrow S_j \cup T_a^j$
      else
        $S_{j+1} \leftarrow S_{j+1} \cup T_a^j$
    $\mathcal{N}^i = \mathcal{N}^i \cup$ `multi-GM`(S_j, S_{j+1})
    $j \leftarrow j + 2^i$.
return $\mathcal{N}^{\log |\chi|}$

Algorithm 2: `multi-GM`(`Set1`, `Set2`)

Input: Two sets `Set1`, `Set2` of subsets of V, from blocks B_j^i and B_{j+1}^i respectively
Output: A set `Fair` of fair merged vertex sets

`Fair` $\leftarrow \varnothing$
while *Set1* $\neq \varnothing$ ***and*** *Set2* $\neq \varnothing$ **do**
  Let $S_1 \in$ `Set1`
  Let $S_2 \in$ `Set2`
  if $|S_1| \geq |S_2|$ **then**
    Let $S \subseteq S_1$ such that $|S| = |S_2|$ and S contains equal number of vertices of each color in B_j^i.
    `Fair` $\leftarrow$ `Fair` $\cup \{S \cup S_2\}$
    $S_1 \leftarrow S_1 \setminus S$
    `Set2` $\leftarrow$ `Set2` $\setminus \{S_2\}$
    if $S_1 = \varnothing$ **then**
      `Set1` $\leftarrow$ `Set1` $\setminus \{S_1\}$
  else
    Let $S \subseteq S_2$ such that $|S| = |S_1|$ and S contains equal number of vertices of each color in B_{j+1}^i
    `Fair` $\leftarrow$ `Fair` $\cup \{S \cup S_1\}$
    $S_2 \leftarrow S_2 \setminus S$
    `Set1` $\leftarrow$ `Set1` $\setminus \{S_1\}$
    if $S_2 = \varnothing$ **then**
      `Set2` $\leftarrow$ `Set2` $\setminus \{S_2\}$
return *Fair*

Closest Fair Clustering for Equi-Proportion: Missing Details

Proof of Lemma 1. To prove this lemma, we need to define the following term

- ith fairness constraint of algorithm `fairpower-of-two`: A clustering $\mathcal{C}$ is said to satisfy the ith fairness constraint of algorithm `fairpower-of-two` if, for every cluster $C \in \mathcal{C}$ and for all pairs of distinct colors $c_k \neq c_\ell$ in the set B_j^i, the number of points of color c_k in C equals the number of points of color c_ℓ in C; that is,

$$c_k(C) = c_\ell(C).$$

Now we prove the following claim

Claim 1. *In the algorithm `fairpower-of-two`, the clustering $\mathcal{N}^i$ is 2-close clustering to $\mathcal{N}^{i-1}$ that satisfies the ith fairness constraint of `fairpower-of-two`, where $\mathcal{N}^0 = \mathcal{D}$.*

We will prove Claim 1 later. First, let us prove Lemma 1 assuming Claim 1.

To prove Lemma 1, we need to prove $\mathcal{N}^{\log |\chi|}$, the output of the algorithm `fairpower-of-two` is $O(|\chi|^{1.6})$-close to $\mathcal{D}$. Let, $\mathcal{N}^*$ be the closest fair clustering to $\mathcal{D}$.

We will prove using mathematical induction that after any iteration i of the algorithm `fairpower-of-two` we have

$$\operatorname{dist}(\mathcal{D}, \mathcal{N}^i) \leq (3^i - 1) \operatorname{dist}(\mathcal{D}, \mathcal{N}^*) \tag{1}$$

For the base case, that is when $i = 1$ by Claim 1 we have $\mathcal{N}^1$ is a 2-close clustering to $\mathcal{D}$ that satisfies the 1st fairness constraint of algorithm `fairpower-of-two`. Since $\mathcal{N}^*$ also satisfies the 1st fairness constraint we have

$$\operatorname{dist}(\mathcal{D}, \mathcal{N}^1) \leq 2 \operatorname{dist}(\mathcal{D}, \mathcal{N}^*)$$

Now, let us assume eq. (1) is true for $i = k - 1$, that is

$$\operatorname{dist}(\mathcal{D}, \mathcal{N}^{k-1}) \leq (3^{k-1} - 1) \operatorname{dist}(\mathcal{D}, \mathcal{N}^*)$$

Now, we prove it for $i = k$.

$$\begin{aligned}
\operatorname{dist}(\mathcal{N}^{k-1}, \mathcal{N}^k) &\leq 2 \operatorname{dist}(\mathcal{N}^{k-1}, \mathcal{N}^*) && (2)\\
&\leq 2(\operatorname{dist}(\mathcal{N}^{k-1}, \mathcal{D}) + \operatorname{dist}(\mathcal{D}, \mathcal{N}^*)) \\
&\quad \text{(triangle inequality)} \\
&\leq 2((3^{k-1} - 1) \operatorname{dist}(\mathcal{D}, \mathcal{N}^*) + \operatorname{dist}(\mathcal{D}, \mathcal{N}^*)) \\
&\quad \text{(by IH)} \\
&= 2 \cdot 3^{k-1} \operatorname{dist}(\mathcal{D}, \mathcal{N}^*) && (3)
\end{aligned}$$

Here, eq. (2) is true because $\mathcal{N}^*$ also satisfies the ith fairness constraint of the algorithm `fairpower-of-two` and due to Claim 1.

Now, we have

$$\begin{aligned}
\operatorname{dist}(\mathcal{D}, \mathcal{N}^k) &\leq \operatorname{dist}(\mathcal{D}, \mathcal{N}^{k-1}) + \operatorname{dist}(\mathcal{N}^{k-1}, \mathcal{N}^k) \\
&\quad \text{(triangle inequality)} \\
&\leq (3^{k-1} - 1) \operatorname{dist}(\mathcal{D}, \mathcal{N}^*) + 2 \cdot 3^{k-1} \operatorname{dist}(\mathcal{D}, \mathcal{N}^*) \\
&\quad \text{(by IH and eq. (3))} \\
&= (3^k - 1) \operatorname{dist}(\mathcal{D}, \mathcal{N}^*) && (4)
\end{aligned}$$

Hence, now we can conclude for $i = \log|\chi|$,

$$\begin{aligned}\operatorname{dist}(\mathcal{D},\mathcal{N}^{\log|\chi|}) &\le (3^{\log|\chi|}-1)\operatorname{dist}(\mathcal{D},\mathcal{N}^*)\\ &= O(|\chi|^{1.6})\operatorname{dist}(\mathcal{D},\mathcal{N}^*)\ (\log_2 3 = 1.6)\end{aligned} \tag{5}$$

Thus the algorithm `fairpower-of-two` computes a $O(\chi^{1.6})$-close *Fair Clustering* which completes the proof of Lemma 1.

Now, let us prove Claim 1. To prove Claim 1, we establish two intermediate claims: Claim 2 and Claim 5.

Let $\mathcal{N}^{i^*}$ denote the closest clustering to $\mathcal{N}^{i-1}$ that satisfies the ith fairness constraint of algorithm `fairpower-of-two`.

In Claim 2, we derive a lower bound on the distance $\operatorname{dist}(\mathcal{N}^{i-1},\mathcal{N}^{i^*})$, and in Claim 5, we provide an upper bound on the distance $\operatorname{dist}(\mathcal{N}^{i-1},\mathcal{N}^{i})$, where $\mathcal{N}^i$ is the clustering obtained after the ith iteration of algorithm `fairpower-of-two`, starting from the initial clustering $\mathcal{D}$.

By comparing the bounds from Claim 2 and Claim 5, we will complete the proof of Claim 1.

To formalize Claim 2, lets recall and introduce some notations:

- Recall, $B_k^{i-1}(N_a^{i-1})$ denotes the set of data points in the cluster $N_a^{i-1} \in \mathcal{N}^{i-1}$ whose colors belong to the block B_k^{i-1}.
- Recall, $s(B_{2k-1}^{i-1}(N_a^{i-1}), B_{2k}^{i-1}(N_a^{i-1})) = T_a^k(say)$ denotes the surplus between B_{2k-1}^{i-1} and B_{2k}^{i-1} in cluster N_a^{i-1}. Specifically,
  - If $|B_{2k-1}^{i-1}(N_a^{i-1})| \ge |B_{2k}^{i}(N_a^{i-1})|$, then

    $$T_a^k \subseteq B_{2k-1}^{i-1}(N_a^{i-1}), \quad |T_a^k| = |B_{2k-1}^{i-1}(N_a^{i-1})| - |B_{2k}^{i-1}(N_a^{i-1})|,$$

    such that $\forall c_j \ne c_\ell \in B_{2k}^{i-1}$, we have $c_j(T_a^k) = c_\ell(T_a^k)$.
  - Otherwise,

    $$T_a^k \subseteq B_{j+1}^{i-1}(N_a^{i-1}), \quad |T_a^k| = |B_{j+1}^{i-1}(N_a^{i-1})| - |B_{j}^{i-1}(N_a^{i-1})|,$$

    such that $\forall c_k \ne c_\ell \in B_{j+1}^{i-1}$, we have $c_k(T_a^k) = c_\ell(T_a^k)$.
- Let, T_a denotes the union of the surpluses within the cluster N_a^{i-1}, computed across all paired color blocks $(B_{2k-1}^{i-1}, B_{2k}^{i-1})$ at iteration $i-1$. More specifically,

$$T_a = \bigcup_{k=1}^{|\chi|/2^i} T_a^k$$

Claim 2.

$$\operatorname{dist}(\mathcal{N}^{i-1},\mathcal{N}^{i^*}) \ge \frac{1}{2}\sum_{N_a^{i-1}\in\mathcal{N}^{i-1}} |T_a|\cdot\left(\left|N_a^{i-1}\right| - |T_a|\right) + |T_a|^2$$

Proof. Consider a cluster $N_a^{i-1} \in \mathcal{N}^{i-1}$. Suppose in $\mathcal{N}^{i^*}$ the cluster N_a^{i-1} is partitioned into $X_1, X_2, \ldots, X_t$, more specifically,

- For all $j \in [t]$, $X_j \subseteq N_{r_j}^{i^*}$ for some $N_{r_j}^{i^*} \in \mathcal{N}^{i^*}$.
- For all $j \ne \ell \in [t]$, we have $N_{r_j}^{i^*} \ne N_{r_\ell}^{i^*}$.
- $\bigcup_{j\in[t]} X_j = N_a^{i-1}$.

By abuse of notation, let us define $B_{2k-1}^{i-1}(X_j)$ and $B_{2k}^{i-1}(X_j)$ be the set of points in X_j that has a color from the blocks B_{2k-1}^{i-1} and B_{2k}^{i-1} respectively.

WLOG assume, $|B_{2k-1}^{i-1}(N_a^{i-1})| > |B_{2k}^{i-1}(N_a^{i-1})|$.

Recall, the blocks are created in such a way that for two consecutive blocks B_{2k-1}^{i-1} and B_{2k}^{i-1}, we have

$$|B_{2k-1}^{i-1}| = |B_{2k}^{i-1}|.$$

Let's create arbitrary pairing of colors $(c, \hat{c})$ where $c \in B_{2k-1}^{i-1}$ and $\hat{c} \in B_{2k}^{i-1}$, we call $\hat{c} \in B_{2k}^{i-1}$ a color corresponding to $c \in B_{2k-1}^{i-1}$.

- **Surplus between two colors $c \in B_{2k-1}^{i-1}$ and $\hat{c} \in B_{2k-1}^{i-1}$ in a cluster** N_a^{i-1} is defined as

  $$s^a(c,\hat{c}) \subseteq B_{2k-1}^{i-1}(N_a^{i-1})$$

  of size $\max(0, c(N_a^{i-1}) - \hat{c}(N_a^{i-1}))$
- **Surplus between two colors $c \in B_{2k-1}^{i-1}$ and $\hat{c} \in B_{2k}^{i-1}$ for a partition** X_j is defined as

  $$\sigma_c^j \subseteq B_{2k-1}^{i-1}(X_j)$$

  of size $\max(0, c(X_j) - \hat{c}(X_j))$ It is straightforward to see that

  $$\sum_{j=1}^{t} |\sigma_c^j| \ge |s^a(c,\hat{c})|$$

Let, us assume $y \in [t]$ be an index such that

$$\sum_{j=1}^{y-1} |\sigma_c^j| < |s^a(c,\hat{c})| \le \sum_{j=1}^{y} |\sigma_c^j|$$

Again assume,

$$\hat{\sigma_c^y} \subseteq \sigma_c^y$$

such that,

$$\sum_{j=1}^{y-1} |\sigma_c^j| + |\hat{\sigma_c^y}| = |s^a(c,\hat{c})|$$

Let us now redefine the notation σ_c^j for $j \in [t]$

$$\sigma_c^j := \begin{cases} \sigma_c^j & \text{if } j < y,\\ \hat{\sigma_c^y} & \text{if } j = y,\\ \varnothing & \text{if } j > y.\end{cases}$$

That is, the sets σ_c^j from $j = 1$ to $(y-1)$ remains unchanged. We shrink the set σ_c^y to include only as many elements as needed to make the summation $|s^a(c,\hat{c})|$. We ignore the sets σ_c^j from $j = (y+1)$ to t completely.

- **Surplus with respect to consecutive pair of blocks B^{i-1}_{2k-1} and B^{i-1}_{2k} for a partition X_j** is defined as

$$S^k_j = \bigcup_{c \in B^{i-1}_{2k-1}} \sigma^j_c$$

- **Surplus with respect to a partition X_j** is defined as

$$S_j = \bigcup_{k=1}^{m_{i-1}} S^k_j$$

where m_{i-1} is the number of blocks created at iteration $(i-1)$.

It is straightforward to see that

$$\sum_{j=1}^{t} |S_j| = T_a$$

Now, since $\mathcal{N}^{i^*}$ satisfies ith fairness constraint of `fairpower-of-two`, hence in $N^{i^*}_{r_j} \in \mathcal{N}^{i^*}$ we have,

$$B^{i-1}_{2k-1}(N^{i^*}_{r_j}) = B^{i-1}_{2k}(N^{i^*}_{r_j}).$$

Recall, $X_j \subseteq N^{i^*}_{r_j}$, hence there must exist at least $|S^k_j|$ vertices having colors from the color block B^{i-1}_{2k} in $N^{i^*}_{r_j}$ that belongs to clusters other than N^{i-1}_a. Let us denote this set of vertices by M^k_j. More specifically,

- $M^k_j \subseteq N^{i^*}_{r_j}$ such that following conditions are satisfied.
  1. $M^k_j \cap N^{i-1}_a = \varnothing$.
  2. $|M^k_j| = |S^k_j|$.
  3. The vertices in M^k_j have colors from the color block B^{i-1}_{2k}.

  Let,

$$M_j = \bigcup_{k=1}^{m_{i-1}} M^k_j$$

Note, since $|M^k_j| = |S^k_j|$ we also have $|M_j| = |S_j|$.

Let us also define

$$M(N^{i-1}_a) = \bigcup_{j=1}^{t} M_j$$

By the definition of $\mathrm{dist}(\mathcal{N}^{i-1}, \mathcal{N}^{i^*})$ is the number of pairs (u, v) that are separated in $\mathcal{N}^{i-1}$ and together in $\mathcal{N}^{i^*}$ or viceversa. Formally, we say that (u, v) are together in $\mathcal{N}^{i^*}$ if there exists $N^{i^*}_k \in \mathcal{N}^{i^*}$ such that $u, v \in N^{i^*}_k$ and we say (u, v) are separated in $\mathcal{N}^{i^*}$ if there exists $N^{i^*}_k, N^{i^*}_j \in \mathcal{N}^{i^*}$ such that $k \neq j$, $u \in N^{i^*}_k$ and $v \in N^{i^*}_j$

Now, to lower bound $\mathrm{dist}(\mathcal{N}^{i-1}, \mathcal{N}^{i^*})$ let us define some costs.

- $\mathrm{cost}_1(N^{i-1}_a)$: For a cluster $N^{i-1}_a \in \mathcal{N}^{i-1}$, $\mathrm{cost}_1(N^{i-1}_a)$ denotes the number of pairs (u, v) such that
  (a) $u \in (N^{i-1}_a \setminus T_a)$ and $v \in T_a$ but separated in $\mathcal{N}^{i^*}$ or,
  (b) $u \in (N^{i-1}_a \setminus T_a$ and $v \in M(N^{i-1}_a)$.
- $\mathrm{cost}_2(N^{i-1}_a)$: For a cluster $N^{i-1}_a \in \mathcal{N}^{i-1}$, $\mathrm{cost}_2(N^{i-1}_a)$ denotes the number of pairs (u, v) such that
  (a) $u, v \in T_a$ but separated in $\mathcal{N}^{i^*}$ or,
  (b) $u \in T_a$ and $v \in M(N^{i-1}_a)$.

We can verify that the pairs counted in $\mathrm{cost}_1(N^{i-1}_a, k)$, and $\mathrm{cost}_2(N^{i-1}_a, k)$ are disjoint and thus we have.

$$\mathrm{dist}(\mathcal{N}^{i-1}, \mathcal{N}^{i^*}) \geq \sum_{N^{i-1}_a \in \mathcal{N}^{i-1}} \frac{1}{2}\mathrm{cost}_1(N^{i-1}_a) + \frac{1}{2}\mathrm{cost}_2(N^{i-1}_a) \tag{6}$$

We multiply $\mathrm{cost}_1(N^{i-1}_a)$ and $\mathrm{cost}_2(N^{i-1}_a)$ with $1/2$ to avoid overcounting of pairs. In both the costs we count the pairs (u, v) in N^{i-1}_a and $M(N^{i-1}_a)$. This pair (u, v) may be counted twice because we take summation overall $N^{i-1}_a \in \mathcal{N}^{i-1}$.

Now, to prove Claim 2, we prove the following

1. $\mathrm{cost}_1(N^{i-1}_a) \geq |T_a|\left(\left|N^{i-1}_a\right| - |T_a|\right)$.
2. $\mathrm{cost}_2(N^{i-1}_a) \geq |T_a|^2$.

It is easy to see that combining eq. (6) and the above statements will prove this Claim 2. So, let us now prove the above statements.

Claim 3. $\mathrm{cost}_1(N^{i-1}_a) \geq |T_a|\left(\left|N^{i-1}_a\right| - |T_a|\right)$

Proof. Recall in N^{i^*}, the cluster N^{i-1}_a is partitioned into $X_1, \ldots, X_t$.

Now, consider a partition X_j of N^{i-1}_a where $j \in [t]$. Let us count the number of pairs (u, v) such that $u \in X_j \setminus S_j$ and $v \in S_\ell$ for $\ell \neq j$. The number of such pairs is

$$|X_j \setminus S_j| \sum_{\ell \neq j} |S_\ell| \tag{7}$$

Let us also count the number of pairs (u, v) such that $u \in X_j \setminus S_j$ and $v \in M_j$. The number of such pairs is

$$\begin{aligned} &|X_j \setminus S_j||M_j| \\ =&|X_j \setminus S_j||S_j| \end{aligned} \tag{8}$$

Now, combining eq. (7) and eq. (8) we get the number of pairs (u, v) such that,

$$(u, v) \in X_j \setminus S_j \times S_\ell \text{ for } \ell \neq j$$

or

$$(u, v) \in X_j \setminus S_j \times M_j$$

is

$$\begin{aligned} &|X_j \setminus S_j| \sum_{\ell \neq j} |S_\ell| \\ &+|X_j \setminus S_j||S_\ell| \\ &=|X_j \setminus S_j|\,|T_a| \end{aligned} \tag{9}$$

Now from eq. (9) we get,

$$\begin{aligned}\text{cost}_1(N_a^{i-1}, k) &\geq \sum_{j=1}^{t} |X_j \setminus S_j| \, |T_a| \\ &= |T_a| \sum_{j=1}^{t} |X_j \setminus S_j| \\ &= |T_a| \left(\left| N_a^{i-1} \right| - |T_a| \right)\end{aligned}$$

□

Claim 4. $\text{cost}_2(N_a^{i-1}) \geq |T_a|^2$

Proof. Consider a partition X_j of N_a^{i-1} where $j \in [t]$. Let us count the number of pairs (u, v) such that $u \in S_j$ and $v \in S_\ell$ for $\ell \neq j$. The number of such pairs is

$$|S_j| \sum_{\ell \neq j} |S_\ell| \tag{10}$$

Let us also count the number of pairs (u, v) such that $u \in S_j$ and $v \in M_j$. The number of such pairs is

$$\begin{aligned}&|S_j||M_j| \\ =&|S_j||S_j|\end{aligned} \tag{11}$$

Now, combining eq. (10) and eq. (11) we get the number of pairs (u, v) such that,

$$(u, v) \in S_j \times S_\ell \text{ for } \ell \neq j$$

or

$$(u, v) \in S_j \times M_j$$

is

$$\begin{aligned}&|S_j| \sum_{\ell \neq j} |S_\ell| + |S_j||S_j| \\ =&\, |S_j| \, |T_a|\end{aligned} \tag{12}$$

Now from eq. (12) we get,

$$\begin{aligned}\text{cost}_2(N_a^{i-1}) &\geq \sum_{j=1}^{t} |S_j| \, |T_a| \\ &= |T_a| \sum_{j=1}^{t} |S_j| \\ &= |T_a| \, |T_a| \\ &= |T_a|^2\end{aligned}$$

□

□

Claim 5.

$$\begin{aligned}\text{dist}(\mathcal{N}^{i-1}, \mathcal{N}^i) \leq &\sum_{N_a^{i-1} \in \mathcal{N}^{i-1}} |T_a| \left(\left| N_a^{i-1} \right| - |T_a| \right) \\ &+ \frac{1}{2} |T_a|^2\end{aligned}$$

Proof. In the algorithm `fairpower-of-two`, from each cluster $N_a^{i-1} \in \mathcal{N}^{i-1}$ we cut the set T_a.

Hence, the pairs (u, v) s.t. $u \in N_a^{i-1} \setminus T_a$ and $v \in T_a$ are counted in $\text{dist}(\mathcal{N}^{i-1}, \mathcal{N}^i)$.

The number of such pairs is

$$|T_a| \left(\left| N_a^{i-1} \right| - |T_a| \right) \tag{13}$$

Again in algorithm 2 the set T_a can further get splitted into multiple subsets $R_1, R_2, \ldots, R_t$ (say). Each of these sets R_i for $i \in [t]$ gets merged with $|R_i|$ points from a different cluster.

Hence, the following pairs (u, v) are counted in $\text{dist}(\mathcal{N}^{i-1}, \mathcal{N}^i)$ which satisfies

1. $u \in R_i$ and v belongs to the set that merged to R_i.
2. $u \in R_i$ and $v \in R_j$ for $i \neq j$.

To count such pairs (u, v) we use a charging scheme, we charge $1/2$ for the vertex u and $1/2$ for the vertex v. That is we define for a set R_i.

$$\begin{aligned}\text{pay}(R_i) = \frac{1}{2} |\{(u, v) \mid &u \in R_i \text{ and } v \in \text{the set merged to } R_i \\ &\text{or } u \in R_i \text{ and } v \in R_j \text{ for } i \neq j\}|\end{aligned}$$

The total number of such pairs is

$$\begin{aligned}\sum_{i=1}^{t} \text{pay}(R_i) &= \sum_{i=1}^{t} \frac{1}{2} |R_i|^2 + \frac{1}{2} |R_i| |(T_a \setminus R_i)| \\ &= \frac{1}{2} |T_a| \sum_{i=1}^{t} |R_i| \\ &= \frac{1}{2} |T_a|^2\end{aligned} \tag{14}$$

By eq. (13) and eq. (14) we get

$$\begin{aligned}\text{dist}(\mathcal{N}^{i-1}, \mathcal{N}^i) \leq &\sum_{N_a^{i-1} \in \mathcal{N}^{i-1}} |T_a| \left(\left| N_a^{i-1} \right| - |T_a| \right) \\ &+ \frac{1}{2} |T_a|^2\end{aligned}$$

□

It is straightforward to see that Claim 2 and Claim 5 proves Claim 1.

□

Proof of Lemma 2. To prove this lemma, let us define the tth fairness constraint of `make-pdc-fair` algorithm.

- tth fairness constraint of `make-pdc-fair` : Let $\mathcal{C}$ be a clustering and let $\{B_k^t\}$ denote the color blocks at the tth iteration in the `make-pdc-fair` algorithm. We say that $\mathcal{C}$ satisfies the tth Fairness Constraint of `make-pdc-fair` if, for every cluster $C_i \in \mathcal{C}$ and every block B_k^t, the color counts satisfy:

$$z_{a_1}(C_i) : z_{a_2}(C_i) : \cdots : z_{a_m}(C_i) = p_{a_1} : p_{a_2} : \cdots : p_{a_m}$$
where $B_k^t = \{z_{a_1}, \ldots, z_{a_m}\}$

To prove Lemma 2, we need to prove the following claim

Claim 6. *For all iterations t of the algorithm `make-pdc-fair`, $\mathcal{F}^t$ is a 6-close clustering to $\mathcal{F}^{t-1}$ that satisfies the tth fairness constraint of `make-pdc-fair`. More specifically,*

$$\operatorname{dist}(\mathcal{F}^{t-1}, \mathcal{F}^t) \leq 6 \operatorname{dist}(\mathcal{F}^{t-1}, \mathcal{F}^{t^*})$$

where $\mathcal{F}^{t^*}$ is the closest clustering to $\mathcal{F}^{t-1}$ that satisfies tth fairness constraint of `make-pdc-fair`.

For now, let us assume Claim 6 and prove Lemma 2.

Proof of Lemma 2: To prove Lemma 2, we need to prove $\mathcal{F}^{\log r}$, the output of the algorithm `make-pdc-fair` is $r^{2.8}$-close to $\mathcal{I}$. Let $\mathcal{F}^*$ be the closest fair clustering to $\mathcal{I}$.

We will prove using mathematical induction that after any iteration t of the algorithm `make-pdc-fair` we have

$$\operatorname{dist}(\mathcal{I}, \mathcal{F}^t) \leq (7^t - 1) \operatorname{dist}(\mathcal{I}, \mathcal{F}^*) \tag{15}$$

For the base case, that is when $t = 1$ by Claim 6 we have $\mathcal{F}^1$ is a 6-close clustering to $\mathcal{I}$ that satisfies the 1st fairness constraint of algorithm `make-pdc-fair`. Since $\mathcal{F}^*$ also satisfies the 1st fairness constraint we have

$$\operatorname{dist}(\mathcal{D}, \mathcal{F}^1) \leq 6 \operatorname{dist}(\mathcal{D}, \mathcal{F}^*)$$

Now, let us assume eq. (15) is true for $t = k - 1$, that is

$$\operatorname{dist}(\mathcal{I}, \mathcal{F}^{k-1}) \leq (7^{k-1} - 1) \operatorname{dist}(\mathcal{I}, \mathcal{F}^*)$$

Now, we prove it for $t = k$.

$$\begin{aligned}
\operatorname{dist}(\mathcal{F}^{k-1}, \mathcal{F}^k) &\leq 6 \operatorname{dist}(\mathcal{F}^{k-1}, \mathcal{F}^*) && (16)\\
&\leq 6(\operatorname{dist}(\mathcal{F}^{k-1}, \mathcal{I}) + \operatorname{dist}(\mathcal{I}, \mathcal{F}^*)) \\
&\quad \text{(by triangle inequality)} \\
&\leq 6 \cdot (7^{k-1} - 1) \operatorname{dist}(I, \mathcal{F}^*) + 6 \operatorname{dist}(I, \mathcal{F}^*) \\
&\quad \text{(by inductive hypothesis)} \\
&= 6 \cdot 7^{k-1} \operatorname{dist}(\mathcal{I}, \mathcal{F}^*) && (17)
\end{aligned}$$

Here, eq. (16) is true because $\mathcal{F}^*$ also satisfies the kth fairness constraint of the algorithm `make-pdc-fair` and due to Claim 6.

Now, we have

$$\begin{aligned}
\operatorname{dist}(\mathcal{I}, \mathcal{F}^k) &\leq \operatorname{dist}(\mathcal{I}, \mathcal{F}^{k-1}) + \operatorname{dist}(\mathcal{F}^{k-1}, \mathcal{F}^k) \\
&\quad \text{(by triangle inequality)} \\
&\leq (7^{k-1} - 1) \operatorname{dist}(\mathcal{I}, \mathcal{F}^*) + 6 \cdot 7^{k-1} \operatorname{dist}(\mathcal{I}, \mathcal{F}^*) \\
&\quad \text{(by inductive hypothesis and eq. (17))} \\
&= (7^k - 1) \operatorname{dist}(\mathcal{I}, \mathcal{F}^*) && (18)
\end{aligned}$$

Hence, now we can conclude for $t = \log r$,

$$\begin{aligned}
\operatorname{dist}(\mathcal{I}, \mathcal{F}^{\log r}) &\leq O(7^{\log r}) \operatorname{dist}(\mathcal{I}, \mathcal{F}^*) \\
&= O(r^{2.8}) \operatorname{dist}(\mathcal{I}, \mathcal{F}^*) \ (\textbf{as } \log_2 7 = 2.8)
\end{aligned} \tag{19}$$

Thus the algorithm `make-pdc-fair` computes a $O(r^{2.8})$-close *Fair Clustering* which completes the proof of Lemma 2.

Now, let us prove Claim 6. To prove Claim 6, we establish four intermediate claims: Claim 7, Claim 8, Claim 9 and Claim 10.

Let $\mathcal{F}^{t^*}$ denote the closest clustering to $\mathcal{F}^{t-1}$ that satisfies the tth fairness constraint of algorithm `make-pdc-fair`.

In Claim 7, Claim 8 and Claim 9 we derive lower bounds on the distance $\operatorname{dist}(\mathcal{F}^{t-1}, \mathcal{F}^{t^*})$, and in Claim 10, we provide an upper bound on the distance $\operatorname{dist}(\mathcal{F}^{t-1}, \mathcal{F}^t)$, where $\mathcal{F}^t$ is the clustering obtained after the tth iteration of algorithm `make-pdc-fair`, starting from the initial clustering $\mathcal{I}$.

By comparing the bounds from Claim 7, Claim 8, Claim 9 and Claim 10, we will complete the proof of Claim 6.

To state Claim 7, we define the surplus and deficit for a cluster $F_a^{t-1} \in \mathcal{F}^{t-1}$.

Recall $\mathcal{F}^{t-1}$ denote the clustering obtained after the $(t-1)$th iteration of the `make-pdc-fair` algorithm and $\{B_1^{t-1}, B_2^{t-1}, \ldots, B_{m_{t-1}}^{t-1}\}$ be the set of color blocks at this iteration. For a color $c_u \in \chi$, recall, p_u denote its proportion in the vertex set V, i.e.,

$$c_1(V) : c_2(V) : \cdots : c_{|\chi|}(V) = p_1 : p_2 : \cdots : p_{|\chi|}.$$

WLOG, we assume $p_1 > p_2 > \cdots > p_{|\chi|}$. Note that due to this assumption for two colors $c_u \in B_j^{t-1}$ and $c_v \in B_k^{t-1}$ where $k > j$ we have $p_v < p_u$.

We define the following for a cluster $F_a^{t-1} \in \mathcal{F}^{t-1}$:

Recall the color blocks B_{2k-1}^{t-1} and B_{2k}^{t-1} are created in such a way that we have $|B_{2k-1}^{t-1}| \geq |B_{2k}^{t-1}|$. Suppose $\tilde{B_{2k-1}^{t-1}} \subseteq B_{2k-1}^{t-1}$ such that $|\tilde{B_{2k-1}^{t-1}}| = |B_{2k}^{t-1}|$.

We create an arbitrary pair of colors $(c_j, c_{\hat{j}})$ where $c_j \in \tilde{B_{2k-1}^{t-1}}$ and $c_{\hat{j}} \in B_{2k}^{t-1}$. We say $c_{\hat{j}} \in B_{2k}^{t-1}$ is a color corresponding to $c_j \in B_{2k-1}^{t-1}$. Note for the colors present in $B_{2k-1}^{t-1} \setminus \tilde{B_{2k-1}^{t-1}}$ we have no corresponding color.

Now we define the following terms

- **Weight of a vertex v:** For $v \in V$, suppose it has a color $c_j \in B_{2k-1}^{t-1}$ and its corresponding color $c_{\hat{j}} \in B_{2k}^{t-1}$. In this case, since $p_j > p_{\hat{j}}$ we define

$$w(v) = \frac{p_{\hat{j}}}{p_j}$$

otherwise if v has color $c_{\hat{j}} \in B_{2k}^{t-1}$ and its corresponding color $c_j \in B_{2k-1}^{t-1}$ then we define

$$w(v) = 1$$

Note, $w(v) \leq 1$ for all $v \in V$.

- **Surplus w.r.t. two corresponding colors c_j and $c_{\hat{j}}$ in a cluster $F_a^{t-1} \in \mathcal{F}^{t-1}$** is defined as

$$s^a(c_j, c_{\hat{j}}) = \max(0, c_{\hat{j}}(F_a^{t-1}) - \frac{p_{\hat{j}}}{p_j} c_j(F_a^{t-1}))$$

- **Surplus w.r.t. consecutive blocks B_{2k-1}^{i-1} and B_{2k}^{i-1} in a cluster $F_a^{t-1} \in \mathcal{F}^{t-1}$** is defined as

$$T_a^k = \sum_{c_j \in B_{2k-1}^{i-1}} s^a(c_j, c_{\hat{j}})$$

- **Surplus of a cluster $F_a^{t-1} \in \mathcal{F}^{t-1}$** is defined as

$$T_a = \sum_{k=1}^{m_{t-1}} T_a^k$$

where m_{t-1} is the number of blocks at iteration $(t-1)$.

- **deficit w.r.t. two corresponding colors c_j and $c_{\hat{j}}$ in a cluster $F_a^{t-1} \in \mathcal{F}^{t-1}$** is defined as

$$d^a(c_j, c_{\hat{j}}) = \max(0, \frac{p_{\hat{j}}}{p_j} c_j(F_a^{t-1}) - c_{\hat{j}}(F_a^{t-1}))$$

- **deficit w.r.t. consecutive blocks B_{2k-1}^{i-1} and B_{2k}^{i-1} in a cluster $F_a^{t-1} \in \mathcal{F}^{t-1}$** is defined as

$$D_a^k = \sum_{c_j \in B_{2k-1}^{i-1}} d^a(c_j, c_{\hat{j}})$$

- **deficit of a cluster $F_a^{t-1} \in \mathcal{F}^{t-1}$** is defined as

$$D_a = \sum_{k=1}^{m_{t-1}} D_a^k$$

Now, we can state Claim 7.

Claim 7.

$$\operatorname{dist}(\mathcal{F}^{t-1}, \mathcal{F}^{t^*}) \geq \frac{1}{4} \sum_{F_a^{t-1} \in \mathcal{F}^{t-1}} T_a \left(\left|F_a^{t-1}\right| - T_a\right) + D_a \left(\left|F_a^{t-1}\right| - T_a\right)$$

Proof. Consider a cluster $F_a^{t-1} \in \mathcal{F}^{t-1}$. Suppose in $\mathcal{F}^{t^*}$ the cluster F_a^{t-1} is partitioned into $X_1, X_2, \ldots, X_s$, more specifically,

- For all $j \in [t]$, $X_j \subseteq F_{r_j}^{t^*}$ for some $F_{r_j}^{t^*} \in \mathcal{F}^{t^*}$.
- For all $j \neq \ell \in [t]$, we have $F_{r_j}^{t^*} \neq F_{r_\ell}^{t^*}$.
- $\bigcup_{j \in [s]} X_j = F_a^{t-1}$.

According to the definition of $\operatorname{dist}(\mathcal{F}^{t-1}, \mathcal{F}^{t^*})$, in $\operatorname{dist}(\mathcal{F}^{t-1}, \mathcal{F}^{t^*})$ we count the number of pairs (u, v) that are present in different clusters in the clustering $\mathcal{F}^{t-1}$ but in the same cluster in the clustering $\mathcal{F}^{t^*}$ or present in the same cluster in the clustering $\mathcal{F}^{t-1}$ but present in different clusters in the clustering $\mathcal{F}^{t^*}$.

To give a lower bound on $\operatorname{dist}(\mathcal{F}^{t-1}, \mathcal{F}^{t^*})$ we calculate the cost incurred by each cluster $F_a^{t-1} \in \mathcal{F}^{t-1}$. Let us describe this formally,

$$\operatorname{pay}(F_a^{t-1}) = |\{(u, v) \mid (u, v \in F_a^{t-1} \wedge u \nsim_{\mathcal{F}^{t^*}} v) \vee (u \in F_a^{t-1}, v \notin F_a^{t-1} \wedge u \sim_{\mathcal{F}^{t^*}} v)\}|$$

where $u \sim_{\mathcal{F}^{t^*}} v$ denotes u and v belongs to the same cluster in the clustering $\mathcal{F}^{t^*}$.

It is straightforward to see that

$$\operatorname{dist}(\mathcal{F}^{t-1}, \mathcal{F}^{t^*}) \geq \frac{1}{2} \sum_{F_a^{t-1} \in \mathcal{F}^{t-1}} \operatorname{pay}(F_a^{t-1}) \tag{20}$$

In the above expression, we multiply by $1/2$ because we took summation overall $F_a^{t-1} \in \mathcal{F}^{t-1}$. To count a pair (u, v) where $u \in F_a^{t-1}$ and $v \in F_b^{t-1}$ for $a \neq b$ we charge $1/2$ when calculating $\operatorname{pay}(F_a^{t-1})$ and again $1/2$ when calculating $\operatorname{pay}(F_b^{t-1})$.

Now, we provide a lower bound on $\operatorname{pay}(F_a^{t-1})$. To provide this lower bound, we need to define some terms.

- **Weight of a color block:**

$$w(B_j^{t-1}) = \sum_{c_u \in B_j^{t-1}} p_u.$$

- **Scaling factor of a color block in cluster F_a^{t-1}:**

$$h(B_j^{t-1}) = \frac{c_u(F_a^{t-1})}{p_u} \quad \text{for any } c_u \in B_j^{t-1},$$

which is well-defined since

$$\frac{c_v(F_a^{t-1})}{p_v} = \frac{c_u(F_a^{t-1})}{p_u} \quad \forall c_u \neq c_v \in B_j^{t-1}.$$

Create two sets `CB` and `MB`.

$$\texttt{CB} = \{(B_i^{t-1}, B_{i+1}^{t-1}) \mid h(B_{i+1}^{t-1}) > h(B_i^{t-1})\}$$

$$\texttt{MB} = \{(B_i^{t-1}, B_{i+1}^{t-1}) \mid h(B_{i+1}^{t-1}) \leq h(B_i^{t-1})\}$$

Informally, the set `CB` contains a pair of color blocks B_i^{t-1} and B_{i+1}^{t+1} if the scaling factor of B_{i+1}^{t+1} is greater than the scaling factor of B_i^{t+1}. In the algorithm `make-pdc-fair`, we cut the surplus for these types of pairs of color blocks. For the pair of color blocks in `MB` we merge the deficit.

Now, for the pairs of color blocks $(B_i^{t+1}, B_{i+1}^{t+1}) \in \texttt{CB}$ let us define some notations w.r.t. a partition X_j.

- Surplus w.r.t. two corresponding colors c_j and $c_{\hat{j}}$ in a partition X_ℓ is defined as

$$\sigma_{c_j}^{\ell} = c_{\hat{j}}(X_\ell) - \frac{p_{\hat{j}}}{p_j} c_j(X_\ell)$$

Here, $(c_j, c_{\hat{j}}) \in B_{2k-1}^{t-1} \times B_{2k}^{t-1}$ where $(B_{2k-1}^{t-1}, B_{2k}^{t-1}) \in \texttt{CB}$.

It is straightforward to see that,

$$s^a(c_j, c_{\hat{j}}) \geq \sum_{\ell=1}^{s} \sigma^{\ell}_{c_j}$$

Similar to the proof of Lemma 1 we can redefine $\sigma^{\ell}_{c_j}$ in such a way that

$$s^a(c_j, c_{\hat{j}}) = \sum_{\ell=1}^{s} \sigma^{\ell}_{c_j}$$

- Surplus w.r.t. two consecutive blocks B^{i-1}_{2k-1} and B^{i-1}_{2k} in a partition X_ℓ is defined as

$$S^k_\ell = \sum_{c_j \in B^{t-1}_{2k-1}} \sigma^{\ell}_{c_j}$$

- Surplus of the partition X_ℓ is

$$S_\ell = \sum_{B^{t-1}_{2k-1} \in \texttt{CB}} S^k_\ell$$

It is straightforward to see that

$$T_a = \sum_{\ell=1}^{s} S_\ell.$$

Since, $\mathcal{F}^{t^*}$ satisfies the tth fairness constraint of the algorithm `make-pdc-fair` we have for a cluster $F^{t^*}_{r_\ell} \in \mathcal{F}^{t^*}$

$$\frac{c_u(F^{t^*}_{r_\ell})}{p_u} = \frac{c_v(F^{t^*}_{r_\ell})}{p_v} \; \forall c_u \in B^{t-1}_{2k-1}, c_v \in B^{t-1}_{2k}$$

recall $X_\ell \subseteq F^{t^*}_{r_\ell} \in \mathcal{F}^{t^*}$.

Let, $M_\ell \subseteq F^{t^*}_{r_\ell} \setminus F^{t-1}_a$

To maintain the tth fairness constraint of the algorithm `make-pdc-fair` we must have

$$|M_\ell| \geq \sum_{v \in M_\ell} w(v) \geq S_\ell$$

Reasons behind the above inequalities

- 1**st inequality:** $w(v) \leq 1$.
- 2**nd inequality:** This follows from the requirement that the cluster $F^{t^*}_{r_\ell}$ must satisfy the tth fairness constraint of the algorithm `make-pdc-fair`. That is, for every pair $(c_j, c_{\hat{j}}) \in B^{t-1}_{2k-1} \times B^{t-1}_{2k}$, we must ensure that after adding the set M_ℓ to X_ℓ in the cluster $F^{t^*}_{r_\ell}$, the following holds:

$$\sum_{v \in C_j(X_\ell \cup M_\ell)} w(v) = \sum_{v \in C_{\hat{j}}(X_\ell \cup M_\ell)} w(v)$$

where $C_r(S)$ denotes the set of vertices of color c_r in a subset $S \subseteq V$. In other words, the total weight of the vertices of color c_j and $c_{\hat{j}}$ in the updated cluster must be equal, for every such pair. The surplus S_ℓ precisely captures the imbalance in these weights in X_ℓ, and hence the total weight added must be at least S_ℓ to restore this balance.

Let, us define $\text{cost}_1(F^{t-1}_a)$ be the number of pairs (u, v) such that

1. Either $u \in M_\ell$ and $v \in X_\ell$
2. or $u \in X_m$ and $v \in X_\ell$ for $m \neq \ell$.

Hence,

$$\begin{aligned}
&\text{cost}_1(F^{t-1}_a) \\
&\geq \frac{1}{2} \sum_{\ell=1}^{s} \left(|M_\ell|(|X_\ell|) + \sum_{m \neq \ell} |X_m|(|X_\ell|) \right) \\
&\geq \frac{1}{2} \sum_{\ell=1}^{s} \left(S_\ell(|X_\ell| - S_\ell) + \sum_{m \neq \ell} S_m(|X_\ell| - S_\ell) \right) \\
&\geq \frac{1}{2} \sum_{\ell=1}^{s} T_a(|X_\ell| - S_\ell) \\
&\geq \frac{1}{2} T_a(|F^{t-1}_a| - T_a) \qquad (21)
\end{aligned}$$

Similarly, now for a pair of color blocks $(B^{t-1}_{2k-1}, B^{t+1}_{2k}) \in \texttt{MB}$ and for a partition X_ℓ, let us define

- $\hat{M}_\ell \subseteq V \setminus F^{t-1}_a$ that serves the deficit for the color blocks B^{t-1}_{2k-1} and B^{t-1}_{2k} in the cluster $F^{t^*}_{r_\ell}$.
- $G_m = \bigcup_{(B^{t-1}_{2k-1}, B^{t-1}_{2k}) \in \texttt{MB}} F^{t^*}_{r_m} \cap B^{t-1}_{2k-1}(F^{t-1}_a)$: it denotes the part of the set $B^{t-1}_{2k-1}(F^{t-1}_a)$ that lies in the cluster $F^{t^*}_{r_m} \in \mathcal{F}^{t^*}$ for $m \neq \ell$

It is straightforward to see that

$$|\hat{M}_\ell| + \sum_{m \neq \ell} |G_m| \geq D_a$$

Let, us define $\text{cost}_2(F^{t-1}_a)$ be the number of pairs (u, v) such that

1. Either $u \in \hat{M}_\ell$ and $v \in X_\ell$
2. or $u \in G_m$ and $v \in X_\ell$ for $m \neq \ell$.

Hence,

$$\begin{aligned}
\text{cost}_2(F^{t-1}_a) &\geq \frac{1}{2} \sum_{\ell=1}^{s} |\hat{M}_\ell||X_\ell| \\
&\quad + \sum_{m \neq \ell} |G_m||X_\ell| \\
&\geq \frac{1}{2} \sum_{j=1}^{s} D_a(|X_\ell| - |S_\ell|) \\
&\geq \frac{1}{2} D_a(|F^{t-1}_a| - T_a) \qquad (22)
\end{aligned}$$

Since, in $\text{cost}_1(F^{t-1}_a)$ and $\text{cost}_2(F^{t-1}_a)$ we count disjoint pairs. Hence,

$$\text{pay}(F^{t-1}_a) \geq \text{cost}_1(F^{t-1}_a) + \text{cost}_2(F^{t-1}_a) \qquad (23)$$

Now. by eq. (20), eq. (21), eq. (22) and eq. (23) we conclude the proof of Claim 7.

□

Claim 8.

$$\text{dist}(\mathcal{F}^{t-1}, \mathcal{F}^{t^*}) \geq \sum_{F_a^{t-1} \in \mathcal{F}^{t-1}} \frac{1}{2} T_a^2$$

Proof. Similar to the proof of the previous claim, we define S_ℓ as the surplus of a partition X_ℓ and $M_\ell \subseteq V \setminus F_a^{t-1}$ be the set of points that are merged to X_ℓ to satisfy the tth fairness constraint of `make-pdc-fair`.

Let, us define $\text{pay}(F_a^{t-1}, X_\ell)$ be the number of pairs (u, v) s.t.

- $u \in M_\ell$, $v \in X_\ell$
- $u \in X_m$, $v \in X_\ell$ for all $m \neq \ell$.

Now,

$$\text{dist}(\mathcal{F}^{t-1}, \mathcal{F}^{t^*}) \geq \frac{1}{2} \sum_{F_a^{t-1} \in \mathcal{F}^{t-1}} \sum_{\ell=1}^{s} \text{pay}(F_a^{t-1}, X_\ell)$$

We multiply by $1/2$ to avoid overcounting of pairs. Note we can overcount a pair in the following situations

(i) Since we take the sum overall $F_a^{t-1} \in \mathcal{F}^{t-1}$, we can overcount a pair (u, v) if $u \in F_b^{t-1}$ and $v \in F_a^{t-1}$ where $b \neq a$ when considering the cluster F_b^{t-1} in the summation.

(ii) Since we take the sum over all partitions X_ℓ of a cluster F_a^{t-1}, we can overcount a pair (u, v) if $u \in X_m$ and $v \in X_\ell$ where $m \neq \ell$ when we consider the partition X_m in the summation.

Now, we only need to show

$$\sum_{\ell=1}^{s} \text{pay}(F_a^{t-1}, X_\ell) \geq T_a^2$$

$$\begin{aligned}
\sum_{\ell=1}^{s} \text{pay}(F_a^{t-1}, X_\ell) &\geq \sum_{\ell=1}^{s} (|M_\ell||X_\ell| + \sum_{m \neq \ell} |X_m||X_\ell|) \\
&\geq \sum_{\ell=1}^{s} (|M_\ell| S_\ell + \sum_{m \neq \ell} S_m S_\ell) \\
&\geq \sum_{\ell=1}^{s} (S_\ell^2 + \sum_{m \neq \ell} S_m S_\ell) \\
&\geq T_a^2
\end{aligned}$$

□

Claim 9.

$$\text{dist}(\mathcal{F}^{t-1}, \mathcal{F}^{t^*}) \geq \sum_{F_a^{t-1} \in \mathcal{F}^{t-1}} \frac{1}{2} D_a^2$$

Proof. For a pair of color blocks $(B_{2k-1}^{t-1}, B_{2k}^{t+1}) \in \texttt{MB}$ and for a partition X_ℓ, let us define

- $\hat{M}_\ell \subseteq V \setminus F_a^{t-1}$ that serves the deficit for the color blocks B_{2k-1}^{t-1} and B_{2k}^{t-1} in the cluster $F_{r_\ell}^{t^*}$.
- $G_m = \bigcup_{(B_{2k-1}^{t-1}, B_{2k}^{t-1}) \in \texttt{MB}} F_{r_m}^{t^*} \cap B_{2k-1}^{t-1}(F_a^{t-1})$: it denotes the part of the set $B_{2k-1}^{t-1}(F_a^{t-1})$ that lies in the cluster $F_{r_m}^{t^*} \in \mathcal{F}^{t^*}$ for $m \neq \ell$

It is straightforward to see that

$$|\hat{M}_\ell| + \sum_{m \neq \ell} |G_m| \geq D_a$$

Let, us define $\text{pay}(F_a^{t-1}, X_\ell)$ be the number of pairs (u, v) s.t.

1. Either $u \in \hat{M}_\ell$ and $v \in X_\ell$
2. or $u \in G_m$ and $v \in X_\ell$ for $m \neq \ell$.

$$\text{dist}(\mathcal{F}^{t-1}, \mathcal{F}^{t^*}) \geq \frac{1}{2} \sum_{F_a^{t-1} \in \mathcal{F}^{t-1}} \sum_{\ell=1}^{s} \text{pay}(F_a^{t-1}, X_\ell)$$

We multiply by $1/2$ to avoid overcounting of pairs. Note, we can overcount a pair in the following situations

(i) Since we take the sum overall $F_a^{t-1} \in \mathcal{F}^{t-1}$, we can overcount a pair (u, v) if $u \in F_b^{t-1}$ and $v \in F_a^{t-1}$ where $b \neq a$ when considering the cluster F_b^{t-1} in the summation.

(ii) Since we take the sum over all partitions X_ℓ of a cluster F_a^{t-1}, we can overcount a pair (u, v) if $u \in G_m$ and $v \in X_\ell$ where $m \neq \ell$ when we consider the partition X_m in the summation. Note $G_m \subseteq X_m$.

Now, we only need to show

$$\sum_{\ell=1}^{s} \text{pay}(F_a^{t-1}, X_\ell) \geq D_a^2$$

$$\begin{aligned}
\sum_{\ell=1}^{s} \text{pay}(F_a^{t-1}, X_\ell) &\geq \sum_{\ell=1}^{s} \left(|\hat{M}_\ell||X_\ell| + \sum_{m \neq \ell} |G_m||X_\ell| \right) \\
&\geq \sum_{\ell=1}^{s} D_a |X_\ell| \\
&\geq D_a |F_a^{t-1}| \\
&\geq D_a^2 \ (\textbf{as} |F_a^{t-1}| \geq D_a)
\end{aligned}$$

□

Claim 10.

$$\begin{aligned}
\text{dist}(\mathcal{F}^{t-1}, \mathcal{F}^{t}) \leq & \sum_{F_a^{t-1} \in \mathcal{F}^{t-1}} T_a \left(\left| F_a^{t-1} \right| - T_a \right) \\
& + D_a \left(\left| F_a^{t-1} \right| - D_a \right) \\
& + \frac{1}{2} D_a^2 + \frac{1}{2} T_a^2
\end{aligned}$$

Proof. For each cluster $F_a^{t-1} \in \mathcal{F}^{t-1}$ we cut the surplus many vertices, T_a and merge deficit many vertices D_a to it.

Hence, in $\text{dist}(\mathcal{F}^{t-1}, \mathcal{F}^{t})$ for a cluster $F_a^{t-1} \in \mathcal{F}^{t-1}$ we count the cost of cutting and merging to it, which is

$$T_a(|F_a^{t-1}| - T_a) + D_a(|F_a^{t-1}| - D_a)$$

Again for a cluster $F_a^{t-1} \in \mathcal{F}^{t-1}$ the deficit many vertices that we merge can come from several other clusters. A trivial upper bound on this cost is given by $(1/2) \cdot D_a^2$.

The surplus many vertices T_a, that we cut from $F_a^{t-1} \in \mathcal{F}^{t-1}$ can further get divided. A trivial upper bound on the cost of dividing T_a many points is $(1/2) \cdot T_a^2$.

Hence we get,

$$\begin{aligned}\operatorname{dist}(\mathcal{F}^{t-1}, \mathcal{F}^t) \leq & \sum_{F_a^{t-1} \in \mathcal{F}^{t-1}} T_a \left(\left|F_a^{t-1}\right| - T_a\right) \\ & + D_a \left(\left|F_a^{t-1}\right| - D_a\right) \\ & + \frac{1}{2} D_a^2 + \frac{1}{2} T_a^2\end{aligned}$$

□

Now we complete the proof of Claim 6.

Proof of Claim 6:. By Claim 10 we get,

$$\begin{aligned}\operatorname{dist}(\mathcal{F}^{t-1}, \mathcal{F}^t) \leq & \sum_{F_a^{t-1} \in \mathcal{F}^{t-1}} T_a \left(\left|F_a^{t-1}\right| - T_a\right) \\ & + D_a \left(\left|F_a^{t-1}\right| - D_a\right) \\ & + \frac{1}{2} D_a^2 + \frac{1}{2} T_a^2 \qquad (24)\end{aligned}$$

By Claim 7 we get

$$\begin{aligned}& \sum_{F_a^{t-1} \in \mathcal{F}^{t-1}} T_a \left(\left|F_a^{t-1}\right| - T_a\right) \\ & + D_a \left(\left|F_a^{t-1}\right| - D_a\right) \leq 4 \operatorname{dist}(\mathcal{F}^{t-1}, \mathcal{F}^{t^*}) \qquad (25)\end{aligned}$$

By Claim 9 we get

$$\sum_{F_a^{t-1} \in \mathcal{F}^{i-1}} \frac{1}{2} D_a^2 \leq \operatorname{dist}(\mathcal{F}^{t-1}, \mathcal{F}^{t^*}) \qquad (26)$$

By Claim 8 we get

$$\sum_{F_a^{t-1} \in \mathcal{F}^{i-1}} \frac{1}{2} T_a^2 \leq \operatorname{dist}(\mathcal{F}^{t-1}, \mathcal{F}^{t^*}) \qquad (27)$$

Now, by combining eq. (24), eq. (25), eq. (26) and eq. (27) we get,

$$\operatorname{dist}(F^{t-1}, F^t) \leq 6 \operatorname{dist}(F^{t-1}, F^{t^*})$$

□

We have shown earlier that Claim 6 implies Lemma 2. Thus, we complete the proof of Lemma 2. □

Proof of Theorem 1. To prove this theorem, let us describe an algorithm `fair-equi`. The algorithm `fair-equi` takes a clustering $\mathcal{D}$ as input. Each vertex of the clustering is colored from the colors $\chi = \{c_1, c_2, \ldots, c_z\}$ where $|\chi|$ is not a power of two.

Describing `fair-equi`: We divide χ into $\log|\chi|$ many disjoint color groups $G_1, G_2, \ldots, G_{\log|\chi|}$ such that the size of each group, $|G_j|$ where $j \in [\log|\chi|]$ is a power of two. We do it greedily according to the binary representation of $|\chi|$. Consider the binary representation of $|\chi|$. For each index $j \in [\log\lceil|\chi|\rceil]$, if the corresponding bit is 1, create a group of size 2^{j-1}.

We apply the algorithm `fairpower-of-two` to get an intermediate clustering $\mathcal{I} = \{I_1, \ldots, I_k\}$ such that for each cluster $I_j \in \mathcal{I}$ where $j \in [k]$ we have for any pair of colors $c_a, c_b \in G_\ell$ where $\ell \in [\log|\chi|]$,

$$c_a(I_j) = c_b(I_j)$$

For a subset $S \subseteq V$, let us define $G_\ell(S)$ as the set of vertices $v \in S$ such that v has a color in G_ℓ. Now, our goal is to get a clustering $\mathcal{F} = \{F_1, \ldots, F_s\}$ from $\mathcal{D}$ such that for each cluster $F_k \in \mathcal{F}$ we have $c_u(F_k) = c_v(F_k)$ where $c_u, c_v \in \chi$.

Given the intermediate clustering $\mathcal{I}$ as input, the algorithm `make-pdc-fair` produces a final *Fair Clustering* $\mathcal{F} = \{F_1, \ldots, F_s\}$ and thus for each cluster $F_j \in \mathcal{F}$, the following proportion holds as stated in Lemma 2:

$$\begin{aligned}& |G_1(F_j)| : |G_2(F_j)| : \ldots : |G_{\log|\chi|}(F_j)| \\ & = |G_1| : |G_2| : \ldots : |G_{\log|\chi|}|\end{aligned}$$

To apply Lemma 2, we consider a group of colors G_ℓ as a single color z_ℓ(say). Here the clustering $\mathcal{I}$ is a p-divisible clustering because for any cluster $I_j \in \mathcal{I}$ we have $|G_\ell(I_j)|$ is divisible by $|G_\ell|$.

Furthermore, the algorithm `make-pdc-fair` ensures uniformity within each group: for every group G_ℓ and for any pair of colors $c_a, c_b \in G_\ell$ with $\ell \in [\log|\chi|]$, it guarantees that $c_a(F_j) = c_b(F_j)$. As a consequence, for any pair of colors $c_u, c_v \in \chi$, we have $c_u(F_j) = c_v(F_j)$, i.e., each color is equally represented within every cluster of $\mathcal{F}$.

By Lemma 2 we get that, $\mathcal{F}$ is $O(\log^{2.8}|\chi|)$-close to the clustering $\mathcal{I}$. By Lemma 1 we get that $\mathcal{I}$ is $O(|\chi|^{1.6})$-close to the clustering $\mathcal{D}$. By applying the triangle inequality to the two preceding results, we conclude that the output clustering $\mathcal{F}$ produced by `fair-equi` is $O(|\chi|^{1.6}\log^{2.8}|\chi|)$-close to the input clustering $\mathcal{D}$.

Let $\mathcal{F}^*$ be the closest fair clustering to $\mathcal{D}$. Hence, we get,

$$\begin{aligned}\operatorname{dist}(\mathcal{D}, \mathcal{F}) & \leq \operatorname{dist}(\mathcal{D}, \mathcal{I}) + \operatorname{dist}(\mathcal{I}, \mathcal{F}) \text{ (triangle inequality)} \\ & \leq \operatorname{dist}(\mathcal{D}, \mathcal{I}) + O(\log^{2.8}|\chi|) \operatorname{dist}(\mathcal{I}, \mathcal{F}^*) \\ & \leq O(|\chi|^{1.6}) \operatorname{dist}(\mathcal{D}, \mathcal{F}^*) + O(\log^{2.8}|\chi|) \operatorname{dist}(\mathcal{I}, \mathcal{F}^*) \\ & \leq O(|\chi|^{1.6}) \operatorname{dist}(\mathcal{D}, \mathcal{F}^*) + O(\log^{2.8}|\chi|)(\operatorname{dist}(\mathcal{D}, \mathcal{I}) \\ & + \operatorname{dist}(\mathcal{D}, \mathcal{F}^*)) \text{ (triangle inequality)} \\ & \leq O(|\chi|^{1.6}\log^{2.8}|\chi| + |\chi| + \log^{2.8}|\chi|) \operatorname{dist}(\mathcal{D}, \mathcal{F}^*) \\ & \leq O(|\chi|^{1.6}\log^{2.8}|\chi|) \operatorname{dist}(\mathcal{D}, \mathcal{F}^*)\end{aligned}$$

This completes the proof of Theorem 1. □

Arbitrary Proportion: Proof of Lemma 3

Proof of Lemma 3. For a color $c_j \in \chi$, we created two sets CUT and MERGE in the algorithm create-pdc. Let us now define two cases.

- Cut case for color c_j: Cut-Case(c_j):

$$\text{If} \sum_{D_k \in \texttt{CUT}} |\sigma(D_k, c_j)| \geq \sum_{D_k \in \texttt{MERGE}} |\mu(D_k, c_j)|$$

- Merge case for color c_j: Merge-Case(c_j):

$$\text{If} \sum_{D_k \in \texttt{CUT}} |\sigma(D_k, c_j)| < \sum_{D_k \in \texttt{MERGE}} |\mu(D_k, c_j)|$$

For Cut-Case(c_j), let us define some costs incurred by our algorithm create-pdc

- From each cluster $D_k \in$ CUT, create-pdc cuts the surplus part $\sigma(D_k, c_j)$ from D_k. Hence, the cost paid for cutting these surplus points is the number of pairs (u, v) such that $u \in \sigma(D_k, c_j)$ and $v \in (D_k \setminus \sigma(D_k, c_j))$. We denote this by:

$$\text{cost}_1(\mathcal{M})^{c_j} = \sum_{D_k \in \texttt{CUT}} |\sigma(D_k, c_j)|(|D_k| - |\sigma(D_k, c_j)|) \tag{28}$$

- For each cluster $D_m \in$ MERGE, the algorithm create-pdc merges the deficit amount $|\delta(D_m, c_j)|$ to these clusters. Hence, the cost paid for merging the deficit to D_m is the number of pairs (u, v) such that $u \in \delta(D_m, c_j)$ and $v \in D_m$

$$\text{cost}_2(\mathcal{M})^{c_j} = \sum_{D_m \in \texttt{MERGE}} |\delta(D_m, c_j)||D_m| \tag{29}$$

- The $\sigma(D_k, c_j)$ points that are cut from D_k can get merged with the surplus of other clusters $\sigma(D_\ell, c_j)$ (say) which are also cut from a cluster $D_\ell \neq D_k$. We call the cost of merging $\sigma(D_k, c_j)$ with the surplus of other clusters as $\text{cost}_3(\mathcal{M})^{c_j}$
  Let, $\sigma(D_k, c_j)$ gets further divided into multiple parts of size $\alpha_1, \ldots, \alpha_t$, so we get,

$$\text{cost}_3(\mathcal{M})^{c_j} \leq \sum_{D_k \in \mathcal{D}} \frac{1}{2} \sum_{i=1}^{t} \alpha_i(p_j - \alpha_i) \tag{30}$$

  In the above expression we provide an upper bound on the number of pairs (u, v) such that $u \in \sigma(D_k, c_x)$ and $v \in V \setminus D_k$ that are present together in a cluster in the clustering $\mathcal{M}$. The above expression provides such an upper bound because of the fact that the surpluses which we cut from the clusters in the CUT in our algorithm is used to fulfil the deficit of a cluster $D_m \in$ MERGE where $m \neq k, \ell$ and we know $\delta(D_m, c_j) < p_j$.
- These $\sigma(D_k, c_j)$ points from D_k can also further be split into several parts $W_1, W_2, \ldots, W_t$ (say). These parts of $\sigma(D_k, c_j)$ points belong to different clusters in $\mathcal{M}$ and thus would incur some cost. We call this cost as $\text{cost}_4(\mathcal{M})^{c_j}$.

$$\text{cost}_4(\mathcal{M})^{c_j} = \sum_{D_k \in \mathcal{D}} \frac{1}{2} \sum_{s=1}^{t} |W_s|(\sigma(D_k, c_j) - |W_s|) \tag{31}$$

  In the above expression, we count the number of pairs (u, v) such that $u, v \in \sigma(D_k, c_j)$ but present in different clusters in the clustering $\mathcal{M}$.

For Merge-Case(c_j), let us define some costs incurred by our algorithm create-pdc.

- In this case for a cluster $D_k \in \mathcal{D}$, we may cut multiple subsets of size p_j and a single subset of size $\sigma(D_k, c_j)$. Let us assume $Y_{k,z}$ denotes the zth such subset of the cluster D_k and $y_{k,z}$ takes the value 1 if we cut zth such subset from D_k. The cost of cutting zth such subset from D_k is given as

$$\kappa_0(D_k) = |\sigma(D_k, c_j)|(|D_k| - |\sigma(D_k, c_j)|)$$
(cost of cutting the 0th subset)
$$\kappa_z(D_k) = p_j(|D_k| - |\sigma(D_k, c_j)| - zp_j)$$
(cost of cutting the zth subset for $z \geq 1$)

  Thus, we define

$$\text{cost}_5(\mathcal{M})^{c_j} = \sum_{D_k \in \mathcal{D}} \sum_{z=0}^{t} y_{k,z}\kappa_z(D_k) \tag{32}$$
$$\text{where } \left(\text{t} = \frac{c_j(D_k) - |\sigma(D_k, c_j)|}{p_j}\right)$$

- Suppose the algorithm create-pdc merges at a cluster $D_m \in$ MERGE$'$. Here, MERGE$' \subseteq$ MERGE denotes the set of clusters where the algorithm create-pdc has merged the deficit amount of points. More specifically, it is defined as

$$D_m \in \texttt{MERGE}' \iff \exists M_\ell \in \mathcal{M}$$
$$\text{s.t. } D_m \subseteq M_\ell$$

  Then, the cost paid for merging the deficit to D_m is

$$\text{cost}_6(\mathcal{M})^{c_j} = \sum_{D_m \in \texttt{MERGE}'} \delta(D_m, c_j)|D_m| \tag{33}$$

- The $|Y_{k,z}|$ points that are cut from D_k can get merged with the subsets $Y_{\ell,z'}$ of some other cluster D_ℓ. We call the cost of merging a subset $Y_{k,z}$ of D_k with the subset $Y_{\ell,z'}$ of another cluster D_ℓ as $\text{cost}_7(M)^{c_j}$.
  Let, $Y_{k,z}$ gets further divided into multiple parts of size $\alpha_1, \ldots, \alpha_t$, so we get,

$$\text{cost}_7(\mathcal{M})^{c_j} \leq \sum_{D_k \in \mathcal{D}} \frac{1}{2} \sum_{i=1}^{t} \alpha_i(p_j - \alpha_i) \tag{34}$$

- The $|Y_{k,z}|$ points that are cut from D_k can also further be split into several parts $W_1, W_2, \ldots, W_t$ (say). These parts of $Y_{k,z}$ can belong to different clusters in $\mathcal{M}$ (output

of `create-pdc`) and thus would incur some cost. We call this cost as $\text{cost}_8(M^{c_j})$.

$$\text{cost}_8(\mathcal{M})^{c_j} = \sum_{D_k \in \mathcal{D}} \frac{1}{2} \sum_{j=1}^{t} |W_j|(|W_{i,z}| - |W_j|) \quad (35)$$

There is a cost which can occur in both the `Cut-Case`(c_j) and `Merge-Case`(c_j).

- Suppose for a cluster $D_k \in \mathcal{D}$, deficit of c_j and another color $c_r \in \chi$ is filled up by the subsets of some other clusters D_ℓ and D_m in $\mathcal{D}$ respectively such that $\ell \neq m$ then this would incur some cost which is the number of pairs (u, v) such that $u \in \delta(D_k, c_j)$ and $v \in \delta(D_k, c_r)$.

$$\text{cost}_9(\mathcal{M})^{c_j} = \sum_{D_k \in D} |\delta(D_k, c_j)||\delta(D_k, c_r)| \quad (36)$$

Let us define $\text{pay}(()\mathcal{M})^{c_j}$ as the number of pairs (u, v) such that at least one of u and v is colored c_j and the following conditions are true.

- u and v are present in the same cluster in $\mathcal{D}$ but in separate clusters in $\mathcal{M}$.
- u and v are present in the separate clusters in $\mathcal{D}$ but in the same cluster in $\mathcal{M}$.

It is straightforward to see that,

$$\text{pay}(\mathcal{M})^{c_j} \leq \sum_{i=1}^{9} \text{cost}_i(\mathcal{M})^{c_j}$$

and

$$\text{dist}(\mathcal{D}, \mathcal{M}) = \sum_{c_j \in \chi} \text{pay}(\mathcal{M})^{c_j} \quad (37)$$

Now, to prove the above Lemma 3, we take the help of the following claims from (Chakraborty et al. 2025a).

Claim 11. *(Chakraborty et al. 2025a)* $\text{cost}_1(\mathcal{M})^{c_j} + \text{cost}_2(\mathcal{M})^{c_j} + \text{cost}_3(\mathcal{M})^{c_j} + \text{cost}_4(\mathcal{M})^{c_j} \leq 3.5\,\text{dist}(\mathcal{D}, \mathcal{M}^*)$

Claim 12. *(Chakraborty et al. 2025a)* $\text{cost}_5(\mathcal{M})^{c_j} + \text{cost}_6(\mathcal{M})^{c_j} + \text{cost}_7(\mathcal{M})^{c_j} + \text{cost}_8(\mathcal{M})^{c_j} \leq 3\,\text{dist}(\mathcal{D}, \mathcal{M}^*)$

Claim 13. *(Chakraborty et al. 2025a)* $\text{cost}_9(\mathcal{M})^{c_j} \leq \text{dist}(\mathcal{D}, \mathcal{M}^*)$

Now we complete the proof of Lemma 3
By Claim 11, Claim 12, Claim 13 and eq. (37) we get

$$\begin{aligned}\text{dist}(\mathcal{D}, \mathcal{M}) &= \sum_{c_j \in \chi} 3.5\,\text{dist}(\mathcal{D}, \mathcal{M}^*) + 3\,\text{dist}(\mathcal{D}, \mathcal{M}^*) \\ &\quad + \text{dist}(\mathcal{D}, \mathcal{M}^*) \\ &= \sum_{c_j \in \chi} 7.5\,\text{dist}(\mathcal{D}, \mathcal{M}^*) \\ &= O(|\chi|)\,\text{dist}(\mathcal{D}, \mathcal{M}^*)\end{aligned}$$

□

Implication to Fair Correlation Clustering: Completing the Proof of Lemma 6

Let us start by recalling the fair correlation clustering problem.

Fair Correlation Clustering. A clustering $\mathcal{F}^*$ is called fair correlation clustering if given a correlation clustering instance G, $\text{cost}(\mathcal{F}^*)$ is minimum among all clusterings $\mathcal{C}$ and it is also a *Fair Clustering*.

β-Approximate Fair Correlation Clustering. A fair clustering $\mathcal{F}$ is called a *β-approximate fair correlation clustering* if:

$$\text{cost}(\mathcal{F}) \leq \beta \cdot \text{cost}(\mathcal{F}^*).$$

Given any arbitrary clustering $\mathcal{C}$ let us construct its corresponding correlation clustering instance $G_\mathcal{C}$ in the following way.

- Let, $G_\mathcal{C}$ be a complete graph that consists of the vertices in $\mathcal{C}$.
- Each edge (u, v) is labelled "+" if u and v are in the same cluster in $\mathcal{C}$.
- (u, v) is labelled "−" if u and v are in different clusters in $\mathcal{C}$.

We know for a correlation clustering instance G, by definition for a clustering $\mathcal{K}$ on G we have

$$\text{cost}(\mathcal{K}) = \text{Total number of intercluster "+" and intracluster "−" edges}$$

Let us now define, for two correlation clustering instances G and H

$$\text{dist}(G, H) = \text{Number of pairs } (u, v) \text{ that are labelled "+" in } G \text{ and "−" in } H \text{ or viceversa.}$$

It is easy to see that for any clustering $\mathcal{K}$,

$$\text{cost}(\mathcal{K}) = \text{dist}(G, G_\mathcal{K}).$$

Let $\mathcal{F}^*$ be the optimal fair correlation clustering of G. We need to prove that

$$\text{dist}(G, G_\mathcal{F}) \leq (\gamma + \beta + \gamma\beta)\,\text{dist}(G, G_{\mathcal{F}^*}).$$

To do this, observe the following:

$$\begin{aligned}\text{dist}(G, G_\mathcal{F}) &\leq \text{dist}(G, G_\mathcal{D}) + \text{dist}(G_\mathcal{D}, G_\mathcal{F}) && (38)\\ &\leq \beta \cdot \text{dist}(G, G_{\mathcal{F}^*}) + \text{dist}(G_\mathcal{D}, G_\mathcal{F}) && (39)\\ &\leq \beta \cdot \text{dist}(G, G_{\mathcal{F}^*}) + \gamma \cdot \text{dist}(G_\mathcal{D}, G_{\mathcal{D}^*}) && (40)\\ &\leq \beta \cdot \text{dist}(G, G_{\mathcal{F}^*}) + \gamma \cdot \text{dist}(G_\mathcal{D}, G_{\mathcal{F}^*}) && (41)\\ &\leq \beta \cdot \text{dist}(G, G_{\mathcal{F}^*}) + \gamma(\text{dist}(G, G_{\mathcal{F}^*}) \\ &\quad + \text{dist}(G, G_\mathcal{D})) && (42)\\ &\leq \beta \cdot \text{dist}(G, G_{\mathcal{F}^*}) + \gamma \cdot \text{dist}(G, G_{\mathcal{F}^*}) \\ &\quad + \gamma\beta \cdot \text{dist}(G, G_{\mathcal{F}^*}) && (43)\\ &= (\gamma + \beta + \gamma\beta) \cdot \text{dist}(G, G_{\mathcal{F}^*})\end{aligned}$$

where:

- (38) follows from the triangle inequality on distance between correlation instances,
- (39) uses the fact that $\mathcal{D}$ is a β-approximation to the optimal fair clustering,
- (40) uses that $\mathcal{F}$ is γ-close to $\mathcal{D}$,
- (41) uses that $\mathcal{F}^*$ is a fair clustering,
- (42) again applies the triangle inequality,
- (43) substitutes the bound from (39).

This completes the proof.

Implication to Fair Consensus Clustering

In this section, we prove the existence of an algorithm that outputs $O(|\chi|^{1.6}\log^{2.8}|\chi|)$-approximate fair consensus clustering in the $(1:1:\cdots:1)$ case and an $O(|\chi|^{3.8})$-approximate fair consensus clustering in the general $(p_1 : p_2 : \cdots : p_{|\chi|})$ case.

Before that, let us formally define Consensus and Fair Consensus Clustering.

Definition 5 (Consensus Clustering)**.** Given a set of clusterings $\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_n$, a clustering $\mathcal{C}^*$ is a consensus clustering if it minimizes the objective

$$\left(\sum_{i=1}^{n} \text{dist}(C_i, \mathcal{C}^*)^\ell\right)^{1/\ell}$$

for any $\ell \in \mathbb{Z}^+$

Definition 6 (Fair Consensus Clustering)**.** Given a set of clusterings $\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_n$, a clustering $\mathcal{F}^*$ is a fair consensus clustering if it minimizes the objective

$$\left(\sum_{i=1}^{n} \text{dist}(C_i, \mathcal{F}^*)^\ell\right)^{1/\ell}$$

and also fair. Here ℓ is any positive integer.

Definition 7 (β-approximate Fair Consensus Clustering)**.** A clustering $\mathcal{F}$ is called a β-approximate Fair Consensus Clustering if the following is true

$$\left(\sum_{i=1}^{n} \text{dist}(C_i, \mathcal{F})^\ell\right)^{1/\ell} \leq \beta \left(\sum_{i=1}^{n} \text{dist}(C_i, \mathcal{F}^*)^\ell\right)^{1/\ell}$$

Now we are ready to state the theorem

Theorem 5. *Given a set of points V, where each $v \in V$ has a color from the set χ. There exists an algorithm that, given a set of clusterings $\mathcal{C}_1, \ldots, \mathcal{C}_m$, finds an $O(|\chi|^{1.6}\log^{2.8}|\chi|)$ approximate consensus fair clustering $\mathcal{F}$ in the $(1:1:\cdots:1)$ case and an $O(|\chi|^{3.8})$ approximate consensus fair clustering in the general $(p_1 : p_2 : \cdots : p_{|\chi|})$ case in $O(m^2|V|^2)$ time.*

To prove the above theorem we need the help of the following lemma, which is implied from an algorithm given by (Chakraborty et al. 2025a).

Lemma 7. *(Chakraborty et al. 2025a) Given a set of points V, where each point $v \in V$ has a color from the set χ. Suppose there exists an algorithm that finds an α-close fair clustering $\mathcal{N}$ to a given clustering $\mathcal{D}$ in $O(|V|\log|V|)$ time, then there exists an algorithm such that given n input clusterings, it finds an $(\alpha+2)$ approximate Fair Consensus Clustering in $O(m^2|V|^2)$ time.*

Proof of Theorem 5. We know that given an input clustering $\mathcal{D}$ the algorithm `fair-equi` outputs an $O(|\chi|^{1.6}\log^{2.8}|\chi|)$ close-fair clustering $\mathcal{F}$ to $\mathcal{D}$ in $(1:1:\cdots:1)$ case and the algorithm `fair-general` outputs $O(|\chi|^{3.8})$ close-fair clustering $\mathcal{F}$ to $\mathcal{D}$ for arbitrary ratios.

Hence, by using Lemma 7 we get that there exists an algorithm that finds a $O(|\chi|^{1.6}\log^{2.8}|\chi|)$ approximate fair consensus clustering in $(1:1:\cdots:1)$ case and $O(|\chi|^{3.8})$ approximate fair consensus clustering for arbitrary ratios in $O(m^2|V|^2)$ time. □

Hardness: Proof of Lemma 4

Lemma 4. *For any integer $k \geq 3$, if S is a* YES *instance of the* 3-PARTITION*, then $(\mathcal{H}, \tau)$ is also a* YES *instance of the* k-CLOSEST EQUIFAIR.

Proof. It suffices to construct a fair clustering $\mathcal{F}$ satisfying $\text{dist}(\mathcal{H}, \mathcal{F}) = \tau$.

Suppose S is a YES instance of the 3-PARTITION. Then there exists a partition $S_1, S_2, \ldots, S_{n/3}$ of $S = \{x_1, x_2, \ldots, x_n\}$ such that for all $1 \leq i \leq n/3$, $|S_i| = 3$ and

$$\sum_{x_j \in S_i} x_j = T \text{ where } T = \frac{\sum_{x_k \in S} x_k}{\frac{n}{3}}.$$

Let, $S_i = \{x_{i_1}, x_{i_2}, x_{i_3}\}$. By our construction of $(\mathcal{H}, \tau)$, we have $\left|\mathrm{R}_{i_j}\right| = x_{i_j}$, for $j \in \{1, 2, 3\}$.

• If $k = 3$, we construct $\mathcal{F}$ by merging R_{i_j} with $\left|\mathrm{R}_{i_j}\right|$ points of color c_2 and $\left|\mathrm{R}_{i_j}\right|$ points of color c_3 in GB_i for $j \in \{1, 2, 3\}$. More formally,

$$\mathcal{F} = \left\{ \left(\mathrm{GB}_{i_j} \cup \mathrm{R}_{i_j}\right) \middle| i \in [n/3], j \in \{1, 2, 3\} \right\}.$$

where $\mathrm{GB}_{i_j} \subseteq \mathrm{GB}_i$ such that GB_{i_j} consists of $\left|\mathrm{R}_{i_j}\right|$ points of color c_2 and $\left|\mathrm{R}_{i_j}\right|$ points of color c_3, for $\in \{1, 2, 3\}$.

It can be seen that $\mathcal{F}$ is a *Fair Clustering* because for each cluster $F = (\mathrm{GB}_{i_j} \cup \mathrm{R}_{i_j}) \in \mathcal{F}$ we have $|c_1(F)| = \left|\mathrm{R}_{i_j}\right| = x_{i_j}$, $|c_2(F)| = \left|c_2(\mathrm{GB}_{i_j})\right| = x_{i_j}$, and $|c_3(F)| = \left|c_3(\mathrm{GB}_{i_j})\right| = x_{i_j}$.

It remains to show that

$$\text{dist}(\mathcal{H}, \mathcal{F}) = 2\sum_{i=1}^{n} x_i^2 + 2\sum_{i=1}^{n} x_i(T - x_i) = \tau.$$

Indeed, for each cluster R_{i_j}, merging with $2\left|\mathrm{R}_{i_j}\right| = 2x_{i_j}$ points from GB_{i_j} costs $2x_{i_j}^2$. Summing this costs for all such clusters results in $2\sum_{i=1}^{n} x_i^2$. Finally, splitting each cluster GB_i into three clusters GB_{i_j} of size $2x_{i_j}$, for $j \in \{1, 2, 3\}$

incurs a cost of $\frac{1}{2}\sum_{j=1}^{3} 2x_{i_j}(2T - 2x_{i_j})$. Summing this costs for all clusters GB_i results in $2\sum_{i=1}^{n} x_i(T - x_i)$.

Hence, $(\mathcal{H}, \tau)$ is a YES instance of the 3-CLOSEST EQUIFAIR.

• If $k \geq 4$, $\mathcal{F}$ is constructed by merging each GB_i with three clusters $\mathrm{R}_{i_1}, \mathrm{R}_{i_2}$, and R_{i_3}. In other words,

$$\mathcal{F} = \{F_i = (\mathrm{GB}_i \cup \mathrm{R}_{i_1} \cup \mathrm{R}_{i_2} \cup \mathrm{R}_{i_3}) \,|\, i \in [n/3]\}.$$

The fairness of $\mathcal{F}$ is ensured since each cluster $F_i \in \mathcal{F}$, $|c_j(F_i)| = |c_j(\mathrm{GB}_i)| = T$, for $2 \leq j \leq k$, and $|c_1(F_i)| = |\mathrm{R}_{i_1}| + |\mathrm{R}_{i_2}| + |\mathrm{R}_{i_3}| = T$.

The distance $\mathrm{dist}(\mathcal{H}, \mathcal{F})$ consists of the following cost. For each GB_i, merging with T points of color c_1 from $\mathrm{R}_{i_1}, \mathrm{R}_{i_2}, \mathrm{R}_{i_3}$ incurs the cost $|\mathrm{GB}_i|\,T = (k-1)T^2$. For each $i = 1, 2, \ldots, n/3$, the cost of merging the three clusters $\mathrm{R}_{i_1}, \mathrm{R}_{i_2}, \mathrm{R}_{i_3}$ together is $\frac{1}{2}\sum_{j=1}^{3} x_{i_j}(T - x_{i_j})$. Overall, we have

$$\mathrm{dist}(\mathcal{H}, \mathcal{F}) = \sum_{i=1}^{n/3}(k-1)T^2 + \frac{1}{2}\sum_{i=1}^{n} x_i(T - x_i) = \tau.$$

This concludes that $(\mathcal{H}, \tau)$ is a YES instance of k-CLOSEST EQUIFAIR. □

Our proof for Lemma 5 utilizes the following result.

Lemma 8 ((Chakraborty et al. 2025a, Lemma 45, Lemma 49)). *Given $S = \{x_1, x_2, \ldots, x_n\}$ a* NO *instance of* 3-PARTITION.

Given an integer $p \geq 2$. Consider a clustering $\mathcal{H}^c = \{\mathrm{B}_1, \mathrm{B}_2, \ldots, \mathrm{B}_{n/3}, \mathrm{R}_1, \mathrm{R}_2, \ldots, \mathrm{R}_n\}$ over a set of red-blue colored points V' where the ratio between the total number of blue and red points is p. In $\mathcal{H}^c$, each B_i is a monochromatic blue cluster of size T, and each R_i is a monochromatic red cluster of size x_i. Let τ be defined as in our reduction with $k = p + 1$, that is,

$$\tau = \begin{cases} \frac{n}{3}\sum_{n=1}^{n/3} pT^2 + \frac{1}{2}\sum_{i=1}^{n} x_i(T - x_i), & \text{if } p \geq 3 \\ 2\sum_{i=1}^{n} x_i^2 + 2\sum_{i=1}^{n} x_i(T - x_i), & \text{if } p = 2 \end{cases}.$$

Then, for every Fair Clustering $\mathcal{F}^c$ over V', it must hold that $\mathrm{dist}(\mathcal{H}^c, \mathcal{F}^c) > \tau$.

Remark 9. Our arguments can be extended to show that the problem of finding a closest fair clustering to a given clustering, under arbitrary color ratios, is also **NP**-complete. The only difference is that we need to adjust the definition of τ in our reduction to account for the arbitrary color ratios. Specifically, if the input clustering $\mathcal{H}$ has color ratios $c_1(V) : c_2(V) : \cdots : c_k(V) = p_1 : p_2 : \cdots : p_k$, where $1 \leq p_1 \leq p_2 \leq \cdots \leq p_k$ are positive integers, then we set

$$\tau = \frac{n}{3}(p_2 + p_3 + \cdots + p_k)\, p_1 T^2 + \frac{1}{2}\sum_{i=1}^{n} p_1^2 x_i\,(T - x_i),$$

if $\frac{p_2+p_3+\cdots+p_k}{p_1} > 1 + \sqrt{2}$, and

$$\tau = \sum_{i=1}^{n} p_1(p_2 + p_3 + \cdots + p_k)x_i^2 + \frac{1}{2}\sum_{i=1}^{n}(p_2 + p_3 + \cdots + p_k)^2 x_i(T - x_i),$$

if $\frac{p_2+p_3+\cdots+p_k}{p_1} < 1 + \sqrt{2}$.

The correctness of this reduction follows analogously to the proofs of Lemma 4 and Lemma 5. We note that in the arguments establishing the mapping from NO instance of 3-PARTITION to a NO instance of k-CLOSEST EQUIFAIR, we employ a variant of Lemma 8, in which the ratio between the number of blue and red points is p/q, with $p > q \geq 1$ being positive integers. The case $p/q > 1 + \sqrt{2}$ is addressed in (Chakraborty et al. 2025a, Remarks 46), while the case $p/q < 1 + \sqrt{2}$ is handled in (Chakraborty et al. 2025a, Remarks 50). This separation explains the two different definitions of τ in our reduction.

Algorithm 3: `make-pdc-fair`

Input: Initial clustering $\mathcal{I} = \mathcal{F}^0$, color set $\zeta = \{z_1, \ldots, z_r\}$, target proportions $(p_1 : \cdots : p_r)$
For each cluster $I_j \in \mathcal{I}$ we have $z_k(I_j)$ is a multiple of p_k.
Output: Clustering $\mathcal{F}$ where each cluster satisfies the global color ratio

Let $T \leftarrow \lceil \log_2 r \rceil$
Initialize blocks $B_j^0 \leftarrow \{z_j\}$ for all $j \in [r]$
Set $\mathcal{F}^0 \leftarrow \mathcal{I}$
for $t \leftarrow 1$ **to** T **do** `// Iterate through block levels`
  Merge adjacent blocks from level $t-1$ to form $\{B_1^t, B_2^t, \ldots\}$
  Let $m_{t-1} \leftarrow$ number of blocks at iteration $t-1$
  if m_{t-1} *is odd* **then**
    Copy the last block as-is: $B^t_{\lceil \#B^{t-1}/2 \rceil} \leftarrow B^{t-1}_{m_{t-1}}$
  Initialize $\mathcal{F}^t \leftarrow \varnothing$
  foreach $F \in \mathcal{F}^{t-1}$ **do** `// Iterate through clusters`
    foreach $B_i^t = B_{2i-1}^{t-1} \cup B_{2i}^{t-1}$ **do** `// Iterate through blocks`
      Let $A = B_{2i-1}^{t-1} = \{z_{a_1}, \ldots, z_{a_s}\}$
      Let $B = B_{2i}^{t-1} = \{z_{b_1}, \ldots, z_{b_u}\}$
      Compute scaling factors:
        $x \leftarrow \min_{j \in [s]} \left(\frac{z_{a_j}(F)}{p_{a_j}} \right)$
        $y \leftarrow \min_{k \in [u]} \left(\frac{z_{b_k}(F)}{p_{b_k}} \right)$
      if $x > y$ **then** `// Case 1: need to merge`
        foreach $k \in [u]$ **do**
          Merge $p_{b_k} \cdot (x - y)$ vertices of color z_{b_k} into F
      else if $x < y$ **then** `// Case 2: need to cut`
        foreach $k \in [u]$ **do**
          Cut $p_{b_k} \cdot (y - x)$ vertices of color z_{b_k} from F
      Add updated cluster F to $\mathcal{F}^t$
return $\mathcal{F}^T$

Algorithm 4: `create-pdc`

Input: Clustering $\mathcal{D}$, colors $\chi = \{c_1, \ldots, c_t\}$, and ratios $p_1 : p_2 : \cdots : p_t$
Output: p-divisible clustering $\mathcal{M}$

foreach $c_j \in \chi$ **do**
  Create σ_j / p_j empty clusters: `extra_clusters` $= \{P_1, P_2, \ldots, P_{\sigma_j/p_j}\}$;
Initialize `CUT` $\leftarrow \varnothing$, `MERGE` $\leftarrow \varnothing$;
foreach $D_i \in \mathcal{D}$ **do**
  if $|\sigma(D_i, c_j)| \leq p_j/2$ **then** `CUT` $\leftarrow$ `CUT` $\cup \{D_i\}$;
  else `MERGE` $\leftarrow$ `MERGE` $\cup \{D_i\}$;
while *CUT* $\neq \varnothing$ **do**
  Pick and remove $D_k \in$ `CUT`;
  Remove surplus: $D_k \leftarrow D_k \setminus \sigma(D_k, c_j)$;
  if *MERGE* $\neq \varnothing$ **then**
    while $\sigma(D_k, c_j) \neq \varnothing$ **do**
      foreach $D_\ell \in$ *MERGE* **do**
        $T \leftarrow$ $\min(|\sigma(D_k, c_j)|, |\delta(D_\ell, c_j)|)$-sized subset of $\sigma(D_k, c_j)$;
        $D_\ell \leftarrow D_\ell \cup T$;
        $\sigma(D_k, c_j) \leftarrow \sigma(D_k, c_j) \setminus T$;
        if $c_j(D_\ell)$ *is a multiple of* p_j **then**
          `MERGE` $\leftarrow$ `MERGE` $\setminus \{D_\ell\}$;
  else
    while $\sigma(D_k, c_j) \neq \varnothing$ **do**
      foreach $P_m \in$ `extra_clusters` **do**
        $Q \leftarrow$ subset of size $\min(p_j, |\sigma(D_k, c_j)|, p_j - |P_m|)$;
        $P_m \leftarrow P_m \cup Q$;
        $\sigma(D_k, c_j) \leftarrow \sigma(D_k, c_j) \setminus Q$;
        if $|P_m| = p_j$ **then**
          `extra_clusters` $\leftarrow$ `extra_clusters` $\setminus \{P_m\}$;
while *MERGE* $\neq \varnothing$ **do**
  Pick $D_k \in$ `CUT` $\cup$ `MERGE` with minimum $\kappa^j(D_k) - \mu^j(D_k)$;
  Remove surplus: $D_k \leftarrow D_k \setminus \sigma(D_k, c_j)$;
  while $\sigma(D_k, c_j) \neq \varnothing$ **do**
    foreach $D_\ell \in$ *MERGE* **do**
      $T \leftarrow \min(|\sigma(D_k, c_j)|, |\delta(D_\ell, c_j)|)$-sized subset;
      $D_\ell \leftarrow D_\ell \cup T$;
      $\sigma(D_k, c_j) \leftarrow \sigma(D_k, c_j) \setminus T$;
      if $c_j(D_\ell)$ *is a multiple of* p_j **then**
        `MERGE` $\leftarrow$ `MERGE` $\setminus \{D_\ell\}$;
return $\mathcal{M}$ *composed of updated* $\mathcal{D}$ *and filled* `extra_clusters`;