

Proportionally Fair Matching with Multiple Groups^{*}

Sayan Bandyapadhyay¹[0000-0001-8875-0102], Fedor Fomin²[0000-0003-1955-4612],
Tanmay Inamdar²[0000-0002-0184-5932], and Kirill Simonov³[0000-0001-9436-7310]

¹ Portland State University, United States sayanb@pdx.edu

² University of Bergen, Norway {Fedor.Fomin,Tanmay.Inamdar}@uib.no

³ Hasso Plattner Institute, University of Potsdam, Germany Kirill.Simonov@hpi.de

Abstract. We study matching problems with the notion of proportional fairness. Proportional fairness is one of the most popular notions of group fairness where every group is represented up to an extent proportional to the final selection size. Matching with proportional fairness or more commonly, proportionally fair matching, was introduced in [Chierichetti et al., AISTATS, 2019]. In this problem, we are given a graph G whose edges are colored with colors from a set C . The task is for given $0 \leq \alpha \leq \beta \leq 1$, to find a maximum (α, β) -balanced matching M in G , that is a matching where for every color $c \in C$ the number of edges in M of color c is between $\alpha|M|$ and $\beta|M|$. Chierichetti et al. initiated the study of this problem with two colors and in the context of bipartite graphs only. However, in many practical applications, the number of colors—although often a small constant—is larger than two. In this work, we make the first step towards understanding the computational complexity of proportionally fair matching with more than two colors. We design exact and approximation algorithms achieving reasonable guarantees on the quality of the matching as well as on the time complexity, and our algorithms work in general graphs. Our algorithms are also supported by suitable hardness bounds.

Keywords: Matching · Fairness · Parameterized Algorithms.

1 Introduction

In this paper, we consider the proportionally fair matching problem in general graphs. Matching is one of the most fundamental notions in graph theory whose study can be traced back to the classical theorems of König [33] and Hall [23]. The first chapter of the book of Lovász and Plummer [38] devoted to matching contains a nice historical overview of the development of the matching problem. The problems of finding a maximum size or a perfect matching are the

^{*} Most of this work was done when all four authors were affiliated with the University of Bergen, Norway. The research leading to these results has received funding from the Research Council of Norway via the project BWCA (grant no. 314528), and the European Research Council (ERC) via grant LOPPRE, reference 819416.

classical algorithmic problems; an incomplete list of references covering the history of algorithmic improvements on these problems is [17,24,29,37,41,42,45,53], see also the book of Schrijver [48] for a historical overview of matching algorithms. Matchings appear naturally in various applications, e.g., kidney transplant matching [47] or numerous assignment problems like assigning products to customers [46]; students to schools [34]; reviewers to manuscripts [7]; and workers to firms [1]. There are scores of works that study fair versions of matchings [9,25,20,27,31,19,51]. Among these distinct notions of matchings, our work is most relevant to the work on (α, β) -balanced matching of Chierichetti et al. [9]. The notion of (α, β) -balanced matching was formulated in [9] by bringing *proportional* fairness and maximum cardinality matching together. Proportional fairness is based on the concept of disparate impact [18], which in the context of matching is defined as follows. A matching is (α, β) -balanced or proportionally fair if the ratio between the number of edges from each group (a color) and the size of the matching is at least α and at most β .

As a motivating example of proportionally fair matching, consider the product recommendation problem in e-commerce. With the advancement of digital marketing and advertising, nowadays companies are interested in more fine-tuned approaches that help them reach the target groups of customers. These groups may be representative of certain underlying demographic categorizations based on gender, age group, geographic location etc. Thus, the number of groups is often a small constant. In particular, in this contemporary setting, one is interested in finding assignments that involve customers from all target groups and have a balanced impact on all these groups. This assignment problem can be modeled as the proportionally fair matching problem between customers and products. In a realistic situation, one might need to assign many products to a customer and many customers to a product. This can be achieved by computing multiple matchings in an iterative manner while removing the edges from the input graph that are already matched.

In a seminal work, Chierichetti et al. [9] obtained a polynomial-time $3/2$ -approximation for the size of the matching, when the input graph is bipartite and the number of groups is 2. However, in many real-world situations, like in the above example, it is natural to assume that the number of target groups is more than 2. Unfortunately, the algorithm of [9] strongly exploits the fact that the number of groups $\ell = 2$. It is not clear how to adapt or extend their algorithm when we have more than two groups. The only known algorithm for $\ell > 2$ groups is an $n^{O(\ell)}$ -time randomized exact algorithm [9,13], which also works for general graphs. The running time of this algorithm has a “bad” exponential dependence on the number of groups, i.e., the running time is not a *fixed* polynomial in n . Thus, this algorithm quickly becomes impractical if ℓ grows. Our research on proportionally fair matching is driven by the following question. *Do there exist efficient algorithms with guaranteed performance for proportionally fair matching when the number of groups ℓ is more than two?*

1.1 Our results and contributions

In this work, we obtain several results on the PROPORTIONALLY FAIR MATCHING problem in general graphs with any arbitrary ℓ number of groups.

- First, we show that the problem is extremely hard for any general ℓ number of groups, in the sense that it is not possible to obtain any approximation algorithm in $2^{o(\ell)}n^{O(1)}$ time even on path graphs, unless the Exponential Time Hypothesis (ETH) [26] is false.
- To complement our hardness result, we design a $1/4\ell$ -approximation algorithm that runs in $2^{O(\ell)}n^{O(1)}$ time. Our algorithm might violate the lower (α) and upper (β) bounds by at most a multiplicative factor of $(1 + 4\ell/|\text{OPT}|)$ if $|\text{OPT}|$ is more than $4\ell^2$, where OPT is any optimum solution. Thus, the violation factor is at most $1 + 1/\ell$, and tends to 1 with asymptotic values of $|\text{OPT}|$.
- We also consider a restricted case of the problem, referred to as the β -limited case in [9], where we only have the upper bound, i.e., no edges might be present from some groups. In this case, we could improve the approximation factor to $1/2\ell$ and running time to polynomial.
- Lastly, we show that the parameterized version of the problem where one seeks for a proportionally fair matching of size k , can be solved exactly in $2^{O(k)}n^{O(1)}$ time. Thus the problem is fixed-parameter tractable parameterized by k .

All of our algorithms are based on simple schemes. Our approximation algorithms use an iterative peeling scheme that in each iteration, extracts a *rainbow* matching containing at most one edge from every group. The exact algorithm is based on a non-trivial application of the celebrated color-coding scheme [2]. These algorithms appear in Sections 3, 4, and 5, respectively. The hardness proof is given in Section 6.

1.2 Related work

In recent years, researchers have introduced and studied several different notions of fairness, e.g., disparate impact [18], statistical parity [52,28], individual fairness [14] and group fairness [15]. Kleinberg et al. [32] formulated three notions of fairness and showed that it is theoretically impossible to satisfy them simultaneously. See also [11,10] for similar exposures.

The notion of proportional fairness with multiple protected groups is widely studied in the literature, which is based on disparate impact [18]. Bei et al. [3] studied the *proportional candidate selection problem*, where the goal is to select a subset of candidates with various attributes from a given set while satisfying certain proportional fairness constraints. Goel et al. [21] considered the problem of learning non-discriminatory and proportionally fair classifiers and proposed the *weighted sum of logs* technique. Proportional fairness has also been considered in the context of Federated learning [55]. Additionally, proportional fairness has been studied in the context of numerous optimization problems including

voting [16], scheduling [30,39], kidney exchange [4,49], and Traveling Salesman Problem [44].

Several different fair matching problems have been studied in the literature. Huang et al. [25] and Boehmer et al. [6] studied fair b -matching, where matching preferences for each vertex are given as ranks, and the goal is to avoid assigning vertices to high ranked preferences as much as possible. Fair-by-design-matching is studied in [20], where instead of a single matching, a probability distribution over all feasible matchings is computed which guarantees individual fairness. See also [27,31].

Apart from the fair versions of matchings, many constrained versions are also studied [50,5]. [50] studied the Bounded Color Matching (BCM) problem where edges are colored and from each color class, only a given number of edges can be chosen. BCM is a special case of 3-set packing and, hence, admits a $3/4$ -approximation [50]. We note that the β -limited case of PROPORTIONALLY FAIR MATCHING is a special case of BCM and, thus, a $3/4$ -approximation follows in this case where the upper bound might be violated by $3/4$ factor. One should compare this factor with our violation factor, which asymptotically tends to 1.

Finally, we note that when the input graph is bipartite, a matching has a natural interpretation as an intersection of two matroids. Matroid intersection has a rich literature containing work on both exact [36,40] and approximation algorithms [35]. However, these algorithms do not consider fairness constraints.

2 Preliminaries

For an integer $\ell \geq 1$, let $[\ell] := \{1, 2, \dots, \ell\}$. Consider any undirected n -vertex graph $G = (V, E)$ such that the edges in E are colored by colors in $C = \{1, \dots, \ell\}$. The function $\chi : E \rightarrow C$ describes the color assignment. For each color $c \in C$, let E_c be the set of edges colored by the color c , i.e., $E_c = \chi^{-1}(c)$. A subset $E' \subseteq E$ is a *matching* in G if no two edges in E' share a common vertex.

Definition 1. *(α, β) -balanced matching.* Given $0 \leq \alpha \leq \beta \leq 1$, a matching $M \subseteq E$ is called (α, β) -balanced if for each color $c \in C$, we have that $\alpha \leq \frac{|M \cap E_c|}{|M|} \leq \beta$.

Thus a matching is (α, β) -balanced if it contains at least α and at most β fraction of edges from every color. In the PROPORTIONALLY FAIR MATCHING problem, the goal is to find a maximum-sized (α, β) -balanced matching. In the restricted β -limited case of the problem, $\alpha = 0$, i.e., we only have the upper bound.

For $\gamma \leq 1$ and $\Delta \geq 1$, a (γ, Δ) -approximation algorithm for PROPORTIONALLY FAIR MATCHING computes a matching of size at least $\gamma \cdot |\text{OPT}|$, where every color appears in at least α/Δ fraction of the edges and in at most $\beta \cdot \Delta$ fraction. OPT is an optimum (α, β) -balanced matching.

A matching is called a *rainbow matching* if all of its edges have distinct colors. We will need the following result due to Gupta et al. [22].

Theorem 1 (Theorem 2 in [22]). *For some integer $k > 0$, suppose there is a rainbow matching in G of size k . There is a $2^k \cdot n^{O(1)}$ time algorithm that computes a rainbow matching of size k .*

3 A $(\frac{1}{4\ell}, 1 + \frac{4\ell}{|OPT|})$ -Approximation for PROPORTIONALLY FAIR MATCHING

In this section, we design an approximation algorithm for PROPORTIONALLY FAIR MATCHING. Let OPT be an optimum (α, β) -balanced matching, $OPT_c = OPT \cap E_c$. We design two algorithms: one for the case when $\alpha > 0$ and the other for the complementary β -limited case. In this section, we slightly abuse the notation, and use OPT (resp. OPT_c for some color $c \in C$) to refer to $|OPT|$ (resp. $|OPT_c|$). The intended meaning should be clear from the context; however we will disambiguate in case there is a possibility of confusion.

First, we consider the $\alpha > 0$ case. Immediately, we have the following observation.

Observation 1 *For any color $c \in C$, OPT contains at least one edge of color c and, hence, G contains a rainbow matching of size ℓ .*

Our algorithm runs in *rounds*. In the following, we define a round. The input in each round is a subgraph $G' = (V', E')$ of G .

Round. Initially $M = \emptyset$. For every color $1 \leq c \leq \ell$, do the following in an iterative manner. If there is no edge of color c in G' , go to the next color or terminate and return (G', M) if $c = \ell$. Otherwise, pick any edge e of color c from G' and add e to the already computed matching M . Remove all the edges (including e) from G' that share a common vertex with e . Repeat the process for the next color with the current (or updated) graph G' or terminate and return (G', M) if $c = \ell$. Thus in each round, we find a rainbow matching in a greedy manner.

Next, we describe our algorithm. The most challenging part of our algorithm is to ensure that the final matching computed is (α, β) -balanced modulo a small factor, i.e., we need to ensure both the lower and the upper bounds within a small factor for each color. Note that just the above greedy way of picking edges might not even ensure that at least one edge is selected from each color. We use the algorithm of [22] in the beginning to overcome this barrier. However, the rest of our algorithm is extremely simple.

The Algorithm. We assume that we know the size of OPT . We describe later how to remove this assumption. Apply the algorithm in Theorem 1 on G to compute a rainbow matching M' of size ℓ . If $OPT \leq 4\ell^2$, return $M := M'$ as the solution and terminate. Otherwise, remove all the edges of M' and the edges adjacent to them from G to obtain the graph G_0 . Initialize M to M' . Greedily pick matched edges in rounds using the **Round** procedure and add them to M until exactly $\lceil OPT/(4\ell) \rceil$ edges are picked in total. In particular, the graph G_0

is the input to the 1-st round and G_1 is the output graph of the 1-st round. G_1 is the input to the 2-nd round and G_2 is the output graph of the 2-nd round, and so on. Note that it might be the case that the last round is not completed fully if the size of M is reached to $\lceil \text{OPT}/(4\ell) \rceil$ before the completion of the round.

Note that the above algorithm is oblivious to α and β in the sense that it never uses these values. Nevertheless, we prove that the computed matching is (α, β) -balanced modulo a small factor. Now we analyze our algorithm.

3.1 The Analysis

Let $M_c = M \cap E_c$. Also, let c^* be a color $c \in C$ such that $|\text{OPT}_c|$ is the minimum at $c = c^*$. The proof of the following observation is fairly straightforward and is in appendix.

Observation 2 $\alpha \leq 1/\ell \leq \beta$.

First we consider the case when $\text{OPT} \leq 4\ell^2$. In this case the returned matching M is a rainbow matching of size exactly ℓ . The existence of such a matching follows by Observation 1. Thus, we immediately obtain a 4ℓ -approximation. As $|M_c|/|M| = 1/\ell$ in this case, by Observation 2, $\alpha \leq |M_c|/|M| \leq \beta$. Thus we obtain the desired result. In the rest of the proof, we analyze the case when $\text{OPT} > 4\ell^2$. We start with the following lemma.

Lemma 1. *The algorithm successfully computes a matching of size exactly $\lceil \text{OPT}/(4\ell) \rceil$. Moreover, for each color c with $\text{OPT}_c > 4\ell$ and round $i \in [1, \lceil \text{OPT}_c/(4\ell) \rceil - 1]$, G_{i-1} contains an edge of color c .*

Proof. Note that by Observation 1, the algorithm in Theorem 1 successfully computes a rainbow matching M' of size ℓ . Now consider any color c such that $\text{OPT}_c \leq 4\ell$. For such a color, M already contains at least $1 \geq \lceil \text{OPT}_c/(4\ell) \rceil$ edge. Now consider any other color c with $|\text{OPT}_c| > 4\ell$. Consider the rainbow matching M' computed in the beginning. As $|M'| = \ell$, the edges of M' can be adjacent to at most 2ℓ edges from OPT , since it is a matching. In particular, the edges of M' can be adjacent to at most 2ℓ edges from the set OPT_c . Hence, G_0 contains at least $\text{OPT}_c - 2\ell$ edges of the set OPT_c . Now consider the execution of round $i \geq 1$. At most ℓ edges are chosen in this round. Hence, these edges can be adjacent to at most 2ℓ edges of OPT_c . It follows that at most 2ℓ fewer edges of the set OPT_c are contained in G_i compared to G_{i-1} . As G_0 has at least $\text{OPT}_c - 2\ell$ edges from the set OPT_c of color c and $\text{OPT}_c > 4\ell$, for each of the first $\lceil (\text{OPT}_c - 2\ell)/(2\ell) \rceil = \lceil \text{OPT}_c/(2\ell) \rceil - 1$ rounds, the algorithm will be able to pick an edge of color c . Thus from such a color c with $\text{OPT}_c > 4\ell$, it can safely pick at least $\lceil \text{OPT}_c/(2\ell) \rceil \geq \lceil \text{OPT}_c/(4\ell) \rceil$ edges in total. Now, as $\text{OPT} = \sum_c \text{OPT}_c$, $\sum_{c \in C} \lceil \text{OPT}_c/(4\ell) \rceil \geq \lceil \text{OPT}/(4\ell) \rceil$. It follows that the algorithm can pick at least $\lceil \text{OPT}/(4\ell) \rceil$ edges. As we stop the algorithm as soon as the size of M reaches to $\lceil \text{OPT}/(4\ell) \rceil$, the lemma follows.

Note that the claimed approximation factor trivially follows from the above lemma. Next, we show that M is (α, β) -balanced modulo a small factor that asymptotically tends to 1 with the size of OPT.

Lemma 2. *For each color $c \in C$, $|M_c| \geq \lceil \text{OPT}_{c^*} / (4\ell) \rceil$.*

Proof. If $\text{OPT}_{c^*} \leq 4\ell$, $|M_c| \geq 1 \geq \text{OPT}_{c^*} / (4\ell)$. So, assume that $\text{OPT}_{c^*} > 4\ell$. Now suppose $|M_c| < \text{OPT}_{c^*} / (4\ell)$ for some c . By Lemma 1, for each of the first $\lceil \text{OPT}_{c^*} / (4\ell) \rceil - 1 \geq \lceil \text{OPT}_{c^*} / (4\ell) \rceil - 1$ rounds, G_{i-1} contains an edge of color c . It follows that the algorithm was forcibly terminated in some round $i \leq (\text{OPT}_{c^*} / (4\ell)) - 1$. Thus, the number of edges chosen from each color $c' \neq c$ is at most $\text{OPT}_{c^*} / (4\ell)$. Hence,

$$|M| = \sum_{c' \neq c} |M_{c'}| + |M_c| < (\ell - 1) \cdot (\text{OPT}_{c^*} / (4\ell)) + (\text{OPT}_{c^*} / (4\ell)) \leq \lceil \text{OPT} / (4\ell) \rceil.$$

This contradicts Lemma 1, which states that we select exactly $\lceil \text{OPT} / (4\ell) \rceil$ edges.

Corollary 1. *For each color $c \in C$, $(|M_c| / |M|) \geq \frac{\alpha}{(1 + 4\ell / \text{OPT})}$.*

Proof. By Lemma 2, $|M_c| \geq \text{OPT}_{c^*} / (4\ell)$.

$$\begin{aligned} \frac{|M_c|}{|M|} &\geq \frac{(\text{OPT}_{c^*} / (4\ell))}{\lceil \text{OPT} / (4\ell) \rceil} \geq \frac{(\text{OPT}_{c^*} / (4\ell))}{(\text{OPT} / (4\ell)) + 1} \\ &= \frac{(\text{OPT}_{c^*}) / (\text{OPT})}{(1 + 4\ell / \text{OPT})} \geq \frac{\alpha}{(1 + 4\ell / \text{OPT})}. \end{aligned}$$

The last inequality follows as OPT satisfies the lower bound for all colors.

Now we turn to proving the upper bound. Let $\alpha^* = \text{OPT}_{c^*} / \text{OPT}$.

Lemma 3. *For each color $c \in C$, $|M_c| \leq \frac{\beta}{\alpha^*} \cdot (\text{OPT}_{c^*} / (4\ell)) + 1$.*

Proof. Suppose for some $c \in C$, $|M_c| > \frac{\beta}{\alpha^*} \cdot (\text{OPT}_{c^*} / (4\ell)) + 1$. Then the number of rounds is strictly greater than $\frac{\beta}{\alpha^*} \cdot (\text{OPT}_{c^*} / (4\ell))$. Now, for any c' , $\text{OPT}_{c'} \geq \alpha^* \cdot \text{OPT}$ and $\text{OPT}_{c'} \leq \beta \cdot \text{OPT}$. Thus, by the definition of α^* , $\frac{\beta}{\alpha^*} \cdot \text{OPT}_{c^*} \geq \text{OPT}_{c'}$. It follows that, for each c' , the number of rounds is strictly greater than $\text{OPT}_{c'} / (4\ell)$. Hence, for each $c' \in C$, more than $(\text{OPT}_{c'} / (4\ell)) + 1$ edges have been chosen. Thus, the total number of edges chosen is strictly larger than

$$\sum_{c' \in C} ((\text{OPT}_{c'} / (4\ell)) + 1) \geq \lceil \text{OPT} / (4\ell) \rceil.$$

This contradicts Lemma 1, which states that we select exactly $\lceil \text{OPT} / (4\ell) \rceil$ edges.

Corollary 2. *For each color $c \in C$, $(|M_c| / |M|) \leq \beta \cdot (1 + \frac{4\ell}{\text{OPT}})$.*

Proof. By Lemma 3,

$$\begin{aligned}
\frac{|M_c|}{|M|} &\leq \frac{(\beta/\alpha^*) \cdot (\text{OPT}_{c^*}/(4\ell)) + 1}{\lceil \text{OPT}/(4\ell) \rceil} \\
&\leq \frac{(\beta/\alpha^*) \cdot (\text{OPT}_{c^*}/(4\ell)) + (\beta/\alpha^*)}{\text{OPT}/(4\ell)} \\
&= \frac{\beta}{\alpha^*} \cdot \frac{\text{OPT}_{c^*}}{\text{OPT}} \cdot \left(1 + \frac{4\ell}{\text{OPT}}\right) \\
&= \frac{\beta}{\alpha^*} \cdot \alpha^* \left(1 + \frac{4\ell}{\text{OPT}}\right) = \beta \cdot \left(1 + \frac{4\ell}{\text{OPT}}\right).
\end{aligned}$$

The second inequality follows, as $\alpha^* \leq \beta$ or $\beta/\alpha^* \geq 1$.

Now let us remove the assumption that we know the size of an optimal solution. Note that $\ell \leq \text{OPT} \leq n$. We probe all values between ℓ and n , and for each such value T run our algorithm. For each matching M returned by the algorithm, we check whether M is $(\frac{\alpha}{(1+4\ell/T)}, \beta \cdot (1 + \frac{4\ell}{T}))$ -balanced. If this is the case, then we keep this solution. Otherwise, we discard the solution. Finally, we select a solution of the largest size among the ones not discarded. By the above analysis, with $T = \text{OPT}$, the matching returned satisfies the desired lower and upper bounds, and has size exactly $\lceil \text{OPT}/(4\ell) \rceil$. Finally, the running time of our algorithm is dominated by $2^\ell n^{O(1)}$ time to compute a rainbow matching algorithm, as stated in Theorem 1.

Theorem 2. *There is a $2^\ell \cdot n^{O(1)}$ time $(1/4\ell, 1 + 4\ell/\text{OPT})$ -approximation algorithm for PROPORTIONALLY FAIR MATCHING with $\alpha > 0$.*

4 A Polynomial-time Approximation in the β -limited Case

In the β -limited case, again we make use of the **Round** procedure. But, the algorithm is slightly different. Most importantly, we do not apply the algorithm in Theorem 1 in the beginning. Thus, our algorithm runs in polynomial time.

The Algorithm. Assume that we know the size of OPT . If $\text{OPT} \leq 2\ell$, we pick any edge and return it as the solution. Otherwise, we just greedily pick matched edges in rounds using the **Round** procedure with the following two cautions. If for a color, at least $\beta \cdot \text{OPT}/(2\ell)$ edges have already been chosen, do not choose any more edge of that color. If at least $(\text{OPT}/2\ell) - 1$ edges have already been chosen, terminate.

Now we analyze the algorithm. First note that if $\text{OPT} \leq 2\ell$, the returned matching has only one edge. The upper bound is trivially satisfied and also we obtain a 2ℓ -approximation. Henceforth, we assume that $\text{OPT} > 2\ell$. Before showing the correctness and analysis of the approximation factor, we show the upper bound for each color. Again let M be the computed matching and $M_c = M \cap E_c$. Later we prove the following lemma in the appendix.

Lemma 4. *Algorithm always returns a matching of size at least $(\text{OPT}/2\ell) - 1$.*

Assuming this we have the following proposition.

Proposition 1. *For each color $c \in C$, $|M_c|/|M| \leq \beta \cdot (1 + \frac{2\ell}{|\text{OPT}|})$.*

Proof. By Lemma 4 and the threshold put on each color in the algorithm, $\frac{|M_c|}{|M|} \leq \frac{\beta \cdot \text{OPT}/(2\ell)}{(\text{OPT}/2\ell) - 1} \leq \beta \cdot (1 + \frac{2\ell}{\text{OPT}})$. The last inequality follows, as $\text{OPT} > 2\ell$.

Theorem 3. *There is a polynomial time algorithm for PROPORTIONALLY FAIR MATCHING in the β -limited case that returns a matching of size at least $(\text{OPT}/2\ell) - 1$ where every color appears in at most $\beta \cdot (1 + 2\ell/\text{OPT})$ fraction of the edges.*

5 An Exact Algorithm for PROPORTIONALLY FAIR MATCHING

Theorem 4. *There is a $2^{O(k)}n^{O(1)}$ -time algorithm that either finds a solution of size k for a PROPORTIONALLY FAIR MATCHING instance, or determines that none exists.*

Proof. We present two different algorithms using the well-known technique of color coding: one for the case $\alpha = 0$ (β -limited case), and one for the case $\alpha > 0$. β -limited case. We aim to reduce the problem to finding a rainbow matching of size k , which we then solve via Theorem 1. The graph G remains the same, however the coloring is going to be different. Namely, for each of the original colors $c \in C$ we color the edges in E_c uniformly and independently at random from a set of k' new colors, where $k' = \lfloor \beta k \rfloor$. Thus, the new instance I' is colored in $\ell \cdot k'$ colors. We use the algorithm of Theorem 1 to find a rainbow matching of size k in the colored graph in I' . Clearly, if a rainbow matching M of size k is found, then the same matching M is a β -limited matching of size k in the original coloring. This holds since by construction for any original color $c \in C$, there are k' new colors in the edge set E_c , and therefore no more than k' edges in $|M \cap E_c|$.

In the other direction, we show that if there exists a β -limited matching M of size k with respect to the original coloring, then with good probability M is a rainbow matching of size k in the new coloring. Assume the original colors c_1, \dots, c_t , for some $1 \leq t \leq \ell$, have non-empty intersection with M , and for each $j \in [t]$ denote $k_j = |M \cap E_{c_j}|$. Observe that $\sum_{j=1}^t k_j = k$, and for each $j \in [t]$, $1 \leq k_j \leq k'$.

Claim. There exists some $\delta > 0$ such that for each $j \in [t]$:

$$\Pr \left[M \cap \left(\bigcup_{i=1}^j E_{c_i} \right) \text{ is a rainbow matching in } I' \right] \geq \exp \left(-\delta \sum_{i=1}^j k_i \right), \quad (1)$$

Proof. We prove the claim by induction on j . For the base case, clearly (1) holds for $j = 0$. Now, fix $j \in [t]$ and assume the statement holds for each $j' < j$, we show that (1) also holds for j . Consider the k_j edges of $M \cap E_{c_j}$, they are colored uniformly and independently in $k' \geq k_j$ colors. By counting possible colorings of $M \cap E_{c_j}$, it follows that

$$\Pr [M \cap E_{c_j} \text{ is a rainbow matching}] \geq \frac{(k')! / (k' - k_j)!}{(k')^{k_j}} \geq \frac{k_j!}{k_j^{k_j}} \geq 2^{-\delta k_j},$$

where the last bound is by Stirling's formula. Now, since colors used for E_{c_j} do not appear anywhere else, using the inductive hypothesis we get

$$\begin{aligned} & \Pr \left[M \cap \left(\bigcup_{i=1}^j E_{c_i} \right) \text{ is a rainbow matching} \right] \\ = & \Pr \left[M \cap \left(\bigcup_{i=1}^{j-1} E_{c_i} \right) \text{ is a rainbow matching} \right] \\ & \cdot \Pr [M \cap E_{c_j} \text{ is a rainbow matching}] \\ \geq & 2^{-\delta \sum_{i=1}^{j-1} k_i} \cdot 2^{-\delta k_j} = 2^{-\delta \sum_{i=1}^j k_i}. \end{aligned}$$

□

Applying (1) with $j = t$, we obtain that M is a rainbow matching with probability at least $2^{-\delta k}$. By repeating the reduction above $2^{O(k)}$ times independently, we achieve that the algorithm succeeds with constant probability.

The case $\alpha > 0$. We observe that in this case, if a matching is fair it necessarily contains at least one edge from each of the groups. Thus, if the number of groups ℓ is greater than k , we immediately conclude there cannot be a fair matching of size k . Otherwise, we guess how the desired k edges are partitioned between the ℓ groups $C = \{c_1, \dots, c_\ell\}$. That is, we guess the numbers k_j for $j \in [\ell]$ such that $\sum_{j=1}^{\ell} k_j = k$, and $\alpha k \leq k_j \leq \beta k$ for each $j \in [\ell]$. From now on, the algorithm is very similar to the β -limited case. For each group c_j , we color the edges of E_{c_j} from a set of k_j colors uniformly and independently at random, where the colors used for each E_{c_j} are non-overlapping. Now we use the algorithm of Theorem 1 to find a rainbow matching of size k . If there is a rainbow matching M of size k , the same matching is a fair matching of size k for the original instance, since in each E_{c_j} exactly k_j edges are chosen, which is at least αk and at most βk . In the other direction, if there is a fair matching M of size k in the original instance, by (1) the matching M is a rainbow matching in the new instance with probability at least $2^{-\delta k}$. Again, by repeating the coloring subprocess independently $2^{O(k)}$ times, we achieve a constant probability of success. Since there are $2^{O(k)}$ options for partitioning k edges into at most $\ell \leq k$ groups, the running time of the whole algorithm is $2^{O(k)} n^{O(1)}$.

Finally, we note that the coloring part in both cases can be derandomized in the standard fashion by using perfect hash families [43], leading to a completely deterministic algorithm.

6 Hardness of Approximation for PROPORTIONALLY FAIR MATCHING

In this section, we show an inapproximability result for PROPORTIONALLY FAIR MATCHING under the Exponential Time Hypothesis (ETH) [26]. ETH states that $2^{\Omega(n)}$ time is needed to solve any generic 3SAT instance with n variables. For our purpose, we need the following restricted version of 3SAT.

3SAT-3

INPUT: Set of clauses $T = \{C_1, \dots, C_m\}$ in variables x_1, \dots, x_n , each clause being the disjunction of 3 or 2 literals, where a literal is a variable x_i or its negation \bar{x}_i . Additionally, each variable appears 3 times.

QUESTION: Is there a truth assignment that simultaneously satisfies all the clauses?

3SAT-3 is known to be NP-hard [54]. We need the following stronger lower bound for 3SAT-3 proved in [12].

Proposition 2 ([12]). *Under ETH, 3SAT-3 cannot be solved in $2^{o(n)}$ time.*

We reduce 3SAT-3 to PROPORTIONALLY FAIR MATCHING which rules out any approximation for the latter problem in $2^{o(\ell)} n^{O(1)}$ time. Our reduction is as follows. For each clause C_i , we have a color i . Also, we have $n - 1$ additional colors $m + 1, \dots, m + n - 1$. Thus, the set of colors $C = \{1, \dots, m + n - 1\}$. For each variable x_i , we construct a gadget, which is a 3-path (a path with 3 edges). Note that x_i can either appear twice in its normal form or in its negated form, as it appears 3 times in total. Let C_{i_1}, C_{i_2} and C_{i_3} be the clauses where x_i appears. Also, suppose it appears in C_{i_1} and C_{i_3} in one form, and in C_{i_2} in the other form. We construct a 3-path P_i for x_i where the j -th edge has color i_j for $1 \leq j \leq 3$. Additionally, we construct $n - 1$ 3-paths $Q_{i,i+1}$ for $1 \leq i \leq n - 1$. All edges of $Q_{i,i+1}$ is of color $m + i$. Finally, we glue together all the paths in the following way to obtain a single path. For each $1 \leq i \leq n - 1$, we glue $Q_{i,i+1}$ in between P_i and P_{i+1} by identifying the last vertex of P_i with the first vertex of $Q_{i,i+1}$ and the last vertex of $Q_{i,i+1}$ with the first vertex of P_{i+1} . Thus we obtain a path P with exactly $3(n + n - 1) = 6n - 3$ edges. Finally, we set $\alpha = \beta = 1/(m + n - 1)$.

Lemma 5. *There is a satisfying assignment for the clauses in 3SAT-3 if and only if there is an (α, β) -balanced matching of size at least $m + n - 1$.*

Proof. Suppose there is a satisfying assignment for all the clauses. For each clause C_j , consider a variable, say x_i , that satisfies C_j . Then there is an edge of color j on P_i . Add this edge to a set M . Thus, after this step, M contains exactly one edge of color j for $1 \leq j \leq m$. Also, note that for each path P_i , if the middle edge is chosen, then no other edge from P_i can be chosen. This is true, as the variable x_i can either satisfy the clauses where it appears in its normal form or the clauses where it appears in its negated form, but not both types of clauses. Hence, M is a matching. Finally, for each path $Q_{i,i+1}$, we add its middle edge to M . Note that M still remains a matching. Moreover, M contains exactly

one edge of color j for $1 \leq j \leq m+n-1$. As $\alpha = \beta = 1/(m+n-1)$, M is an (α, β) -balanced matching.

Now suppose there is an (α, β) -balanced matching M of size at least $m+n-1$. First, we show that $|M| = m+n-1$. Note that if $|M| > m+n-1$, then the only possibility is that $|M| = 2(m+n-1)$, as $\alpha = \beta$ and at most 2 edges of color j can be picked in any matching for $m+1 \leq j \leq m+n-1$. Suppose $|M| = 2(m+n-1)$. Then from each $Q_{i,i+1}$, M contains the first and the third edge. This implies, from each P_t , $1 \leq t \leq n$, we can pick at most one edge. Thus, total number of edges in M is at most $2(n-1) + n$. It follows that $2m+2n-2 \leq 2n-2+n$ or $n \geq 2m$. Now, in 3SAT-3 the total number of literals is $3n$ and at most $3m$, as each variable appears 3 times and each clause contains at most 3 literals. This implies $n \leq m$, and we obtain a contradiction. Thus, $|M| = m+n-1$. Now, consider any P_i . In the first case, the first and third edges of P_i are corresponding to literal x_i and, hence, the middle edge is corresponding to the literal \bar{x}_i . If the middle edge is in M , assign 0 to x_i , otherwise, assign 1 to x_i . In the other case, if the middle edge is in M , assign 1 to x_i , otherwise, assign 0 to x_i . We claim that the constructed assignment satisfies all the clauses. Consider any clause C_j . Let $e \in P_i$ be the edge in M of color j for $1 \leq j \leq m$. Note that e can be the middle edge in P_i or not. In any case, if e is corresponding to \bar{x}_i , we assigned 0 to x_i , and if e is corresponding to x_i , we assigned 1 to x_i . Thus, in either case, C_j is satisfied. This completes the proof of the lemma. □

Note that for a 3SAT-3 instance the total numbers of literals is $3n$. As each clause contains at least 2 literals, $m \leq 3n/2$. Now, for the instances constructed in the above proof, the number of colors $\ell = m+n-1 \leq 3n/2+n-1 = 5n/2-1$. Thus, the above lemma along with Proposition 2 show that it is not possible to decide whether there is an (α, β) -balanced matching of a given size in time $2^{o(\ell)}n^{O(1)}$. Using this, we also show that even no $2^{o(\ell)}n^{O(1)}$ time approximation algorithm is possible. Suppose there is a $2^{o(\ell)}n^{O(1)}$ time γ -approximation algorithm, where $\gamma < 1$. For our constructed path instances, we apply this algorithm to obtain a matching. Note that the γ -approximate solution M must contain at least one edge of every color, as $\alpha = \beta$. By the proof in the above lemma, $|M|$ is exactly $m+n-1$. Hence, using this algorithm, we can decide in $2^{o(\ell)}n^{O(1)}$ time whether there is an (α, β) -balanced matching of size $m+n-1$. But, this is a contradiction, which leads to the following theorem.

Theorem 5. *For any $\gamma > 1$, under ETH, there is no $2^{o(\ell)}n^{O(1)}$ time γ -approximation algorithm for PROPORTIONALLY FAIR MATCHING, even on paths.*

7 Conclusions

In this paper, we study the notion of proportional fairness in the context of matchings in graphs, which has been studied by Chierichetti et al. [8]. We obtained approximation and exact algorithms for the proportionally fair matching problem. We also complement these results by showing hardness results. It

would be interesting to obtain a $o(\ell)$ - or a true $O(\ell)$ -approximation for PROPORTIONALLY FAIR MATCHING improving our result. As evident from our hardness result, there is a lower bound of $2^{\Omega(\ell)}n^{O(1)}$ on the running time of such an algorithm.

References

1. Ahmadi, S., Ahmed, F., Dickerson, J.P., Fuge, M., Khuller, S.: An algorithm for multi-attribute diverse matching. In: Bessiere, C. (ed.) Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI 2020. pp. 3–9. [ijcai.org \(2020\)](https://ijcai.org/2020) 2
2. Alon, N., Yuster, R., Zwick, U.: Color-coding. *Journal of the ACM (JACM)* **42**(4), 844–856 (1995) 3
3. Bei, X., Liu, S., Poon, C.K., Wang, H.: Candidate selections with proportional fairness constraints. In: Seghrouchni, A.E.F., Sukthankar, G., An, B., Yorke-Smith, N. (eds.) Proceedings of the 19th International Conference on Autonomous Agents and Multiagent Systems, AAMAS '20, Auckland, New Zealand, May 9-13, 2020. pp. 150–158. International Foundation for Autonomous Agents and Multiagent Systems (2020). <https://doi.org/10.5555/3398761.3398784>, <https://dl.acm.org/doi/10.5555/3398761.3398784> 3
4. Benedek, M., Biró, P., Kern, W., Paulusma, D.: Computing balanced solutions for large international kidney exchange schemes. In: Faliszewski, P., Mascardi, V., Pelachaud, C., Taylor, M.E. (eds.) 21st International Conference on Autonomous Agents and Multiagent Systems, AAMAS 2022, Auckland, New Zealand, May 9-13, 2022. pp. 82–90. International Foundation for Autonomous Agents and Multiagent Systems (IFAAMAS) (2022). <https://doi.org/10.5555/3535850.3535861>, <https://www.ifaamas.org/Proceedings/aamas2022/pdfs/p82.pdf> 4
5. Berger, A., Bonifaci, V., Grandoni, F., Schäfer, G.: Budgeted matching and budgeted matroid intersection via the gasoline puzzle. *Math. Program.* **128**(1-2), 355–372 (2011) 4
6. Boehmer, N., Koana, T.: The complexity of finding fair many-to-one matchings. In: Bojanczyk, M., Merelli, E., Woodruff, D.P. (eds.) 49th International Colloquium on Automata, Languages, and Programming, ICALP 2022, July 4-8, 2022, Paris, France. LIPIcs, vol. 229, pp. 27:1–27:18. Schloss Dagstuhl - Leibniz-Zentrum für Informatik (2022). <https://doi.org/10.4230/LIPIcs.ICALP.2022.27>, <https://doi.org/10.4230/LIPIcs.ICALP.2022.27> 4
7. Charlin, L., Zemel, R.: The toronto paper matching system: an automated paper-reviewer assignment system (2013) 2
8. Chierichetti, F., Kumar, R., Lattanzi, S., Vassilvitskii, S.: Fair clustering through fairlets. In: Advances in Neural Information Processing Systems. pp. 5029–5037 (2017) 12
9. Chierichetti, F., Kumar, R., Lattanzi, S., Vassilvitskii, S.: Matroids, matchings, and fairness. In: Chaudhuri, K., Sugiyama, M. (eds.) The 22nd International Conference on Artificial Intelligence and Statistics, AISTATS 2019, 16-18 April 2019, Naha, Okinawa, Japan. Proceedings of Machine Learning Research, vol. 89, pp. 2212–2220. PMLR (2019) 2, 3
10. Chouldechova, A.: Fair prediction with disparate impact: A study of bias in recidivism prediction instruments. *Big data* **5**(2), 153–163 (2017) 3

11. Corbett-Davies, S., Pierson, E., Feller, A., Goel, S., Huq, A.: Algorithmic decision making and the cost of fairness. In: Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Halifax, NS, Canada, August 13 - 17, 2017. pp. 797–806. ACM (2017) [3](#)
12. Cygan, M., Marx, D., Pilipczuk, M., Pilipczuk, M.: Hitting forbidden subgraphs in graphs of bounded treewidth. *Information and Computation* **256**, 62–82 (2017) [11](#)
13. Czabarka, E., Szekely, L.A., Toroczkai, Z., Walker, S.: An algebraic monte-carlo algorithm for the bipartite partition adjacency matrix realization problem. arXiv preprint arXiv:1708.08242 (2017) [2](#)
14. Dwork, C., Hardt, M., Pitassi, T., Reingold, O., Zemel, R.: Fairness through awareness. In: Proceedings of the 3rd innovations in theoretical computer science conference. pp. 214–226 (2012) [3](#)
15. Dwork, C., Ilvento, C.: Group fairness under composition. In: Proceedings of the 2018 Conference on Fairness, Accountability, and Transparency (FAT* 2018) (2018) [3](#)
16. Ebadian, S., Kahng, A., Peters, D., Shah, N.: Optimized distortion and proportional fairness in voting. In: Proceedings of the 23rd ACM Conference on Economics and Computation. pp. 563–600 (2022) [4](#)
17. Edmonds, J.: Paths, trees, and flowers. *Can. J. Math.* **17**(3), 449–467 (1965) [2](#)
18. Feldman, M., Friedler, S.A., Moeller, J., Scheidegger, C., Venkatasubramanian, S.: Certifying and removing disparate impact. In: proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining. pp. 259–268 (2015) [2](#), [3](#)
19. Freeman, R., Micha, E., Shah, N.: Two-sided matching meets fair division. In: Zhou, Z. (ed.) Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI 2021, Virtual Event / Montreal, Canada, 19-27 August 2021. pp. 203–209. ijcai.org (2021). <https://doi.org/10.24963/ijcai.2021/29>, <https://doi.org/10.24963/ijcai.2021/29> [2](#)
20. García-Soriano, D., Bonchi, F.: Fair-by-design matching. *Data Min. Knowl. Discov.* **34**(5), 1291–1335 (2020) [2](#), [4](#)
21. Goel, N., Yaghini, M., Faltings, B.: Non-discriminatory machine learning through convex fairness criteria. In: Furman, J., Marchant, G.E., Price, H., Rossi, F. (eds.) Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society, AIES 2018, New Orleans, LA, USA, February 02-03, 2018. p. 116. ACM (2018). <https://doi.org/10.1145/3278721.3278722>, <https://doi.org/10.1145/3278721.3278722> [3](#)
22. Gupta, S., Roy, S., Saurabh, S., Zehavi, M.: Parameterized algorithms and kernels for rainbow matching. *Algorithmica* **81**(4), 1684–1698 (2019) [4](#), [5](#)
23. Hall, P.: On representatives of subsets. *J. London Math. Soc.* **10**, 26–30 (1935) [1](#)
24. Hopcroft, J.E., Karp, R.M.: An $n^{5/2}$ algorithm for maximum matchings in bipartite graphs. *SIAM Journal on Computing* **2**, 225–231 (1973) [2](#)
25. Huang, C., Kavitha, T., Mehlhorn, K., Michail, D.: Fair matchings and related problems. *Algorithmica* **74**(3), 1184–1203 (2016) [2](#), [4](#)
26. Impagliazzo, R., Paturi, R.: On the complexity of k-sat. *Journal of Computer and System Sciences* **62**(2), 367–375 (2001) [3](#), [11](#)
27. Kamada, Y., Kojima, F.: Fair matching under constraints: Theory and applications (2020) [2](#), [4](#)
28. Kamishima, T., Akaho, S., Sakuma, J.: Fairness-aware learning through regularization approach. In: Spiliopoulou, M., Wang, H., Cook, D.J., Pei, J., Wang, W.,

- Zaïane, O.R., Wu, X. (eds.) Data Mining Workshops (ICDMW), 2011 IEEE 11th International Conference on, Vancouver, BC, Canada, December 11, 2011. pp. 643–650. IEEE Computer Society (2011) 3
29. Karzanov, A.V.: The problem of finding the maximal flow in a network by the method of preflows. Dokl. Akad. Nauk SSSR **215**, 49–52 (1974) 2
 30. Kesavan, D., Periyathambi, E., Chokkalingam, A.: A proportional fair scheduling strategy using multiobjective gradient-based african buffalo optimization algorithm for effective resource allocation and interference minimization. International Journal of Communication Systems **35**(1), e5003 (2022) 4
 31. Klaus, B., Klijn, F.: Procedurally fair and stable matching. Economic Theory **27**(2), 431–447 (2006) 2, 4
 32. Kleinberg, J., Mullainathan, S., Raghavan, M.: Inherent trade-offs in the fair determination of risk scores. In: 8th Innovations in Theoretical Computer Science Conference (ITCS 2017). Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik (2017) 3
 33. König, D.: Über Graphen und ihre Anwendung auf Determinantentheorie und Mengenlehre. Math. Ann. **77**(4), 453–465 (1916) 1
 34. Kurata, R., Hamada, N., Iwasaki, A., Yokoo, M.: Controlled school choice with soft bounds and overlapping types. Journal of Artificial Intelligence Research **58**, 153–184 (2017) 2
 35. Linhares, A., Olver, N., Swamy, C., Zenklusen, R.: Approximate multi-matroid intersection via iterative refinement. Math. Program. **183**(1), 397–418 (2020). <https://doi.org/10.1007/s10107-020-01524-y>, <https://doi.org/10.1007/s10107-020-01524-y> 4
 36. Lokshтанov, D., Misra, P., Panolan, F., Saurabh, S.: Deterministic truncation of linear matroids. ACM Trans. Algorithms **14**(2), 14:1–14:20 (2018). <https://doi.org/10.1145/3170444>, <https://doi.org/10.1145/3170444> 4
 37. Lovász, L.: On determinants, matchings, and random algorithms. In: Fundamentals of Computation Theory. pp. 565–574. Akademie Verlag (1979) 2
 38. Lovász, L., Plummer, M.D.: Matching Theory. AMS (2009) 1
 39. Lu, Y.: The optimization of automated container terminal scheduling based on proportional fair priority. Mathematical Problems in Engineering **2022** (2022) 4
 40. Marx, D.: A parameterized view on matroid optimization problems. Theoretical Computer Science **410**(44), 4471–4479 (2009) 4
 41. Mađry, A.: Navigating central path with electrical flows: From flows to matchings, and back. In: FOCS 2013. pp. 253–262. IEEE Computer Society (2013) 2
 42. Mucha, M., Sankowski, P.: Maximum matchings via Gaussian elimination. In: FOCS 2004. pp. 248–255. IEEE Computer Society (2004) 2
 43. Naor, M., Schulman, L.J., Srinivasan, A.: Splitters and near-optimal derandomization. In: Proceedings of the 36th Annual Symposium on Foundations of Computer Science (FOCS 1995). pp. 182–191. IEEE (1995) 10
 44. Nguyen, M.H., Baiou, M., Nguyen, V.H., Vo, T.Q.T.: Nash fairness solutions for balanced tsp. In: 10th International Network Optimization Conference (INOC) (2022) 4
 45. Rabin, M.O., Vazirani, V.V.: Maximum matchings in general graphs through randomization. J. Algorithms **10**(4), 557–567 (1989) 2
 46. Ristoski, P., Petrovski, P., Mika, P., Paulheim, H.: A machine learning approach for product matching and categorization. Semantic web **9**(5), 707–728 (2018) 2
 47. Roth, A.E., Sönmez, T., Ünver, M.U.: Efficient kidney exchange: Coincidence of wants in markets with compatibility-based preferences. American Economic Review **97**(3), 828–851 (2007) 2

48. Schrijver, A.: Combinatorial Optimization. Polyhedra and Efficiency. Vol. A. Springer-Verlag, Berlin (2003) [2](#)
49. St-Arnaud, W., Carvalho, M., Farnadi, G.: Adaptation, comparison and practical implementation of fairness schemes in kidney exchange programs. arXiv preprint arXiv:2207.00241 (2022) [4](#)
50. Stamoulis, G.: Approximation algorithms for bounded color matchings via convex decompositions. In: Csuhaj-Varjú, E., Dietzfelbinger, M., Ésik, Z. (eds.) Mathematical Foundations of Computer Science 2014 - 39th International Symposium, MFCS 2014, Budapest, Hungary, August 25-29, 2014. Proceedings, Part II. Lecture Notes in Computer Science, vol. 8635, pp. 625–636. Springer (2014) [4](#)
51. Sun, Z., Todo, T., Walsh, T.: Fair pairwise exchange among groups. In: IJCAI. pp. 419–425 (2021) [2](#)
52. Thanh, B.L., Ruggieri, S., Turini, F.: k-nn as an implementation of situation testing for discrimination discovery and prevention. In: Apté, C., Ghosh, J., Smyth, P. (eds.) Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Diego, CA, USA, August 21-24, 2011. pp. 502–510. ACM (2011) [3](#)
53. Vazirani, V.V.: A proof of the MV matching algorithm (2014) [2](#)
54. Yannakakis, M.: Node- and edge-deletion np-complete problems. In: Lipton, R.J., Burkhard, W.A., Savitch, W.J., Friedman, E.P., Aho, A.V. (eds.) Proceedings of the 10th Annual ACM Symposium on Theory of Computing, May 1-3, 1978, San Diego, California, USA. pp. 253–264. ACM (1978) [11](#)
55. Zhang, G., Malekmohammadi, S., Chen, X., Yu, Y.: Equality is not equity: Proportional fairness in federated learning. arXiv preprint arXiv:2202.01666 (2022) [3](#)

A Missing Proofs

Proof (of Observation 2). Let \hat{c} be a color $c \in C$ such that $|\text{OPT}_c|$ is the maximum at $c = \hat{c}$. By definition, $\text{OPT} \geq \ell \cdot \text{OPT}_{c^*}$, or $\text{OPT}_{c^*}/\text{OPT} \leq 1/\ell$. Thus, $\alpha \leq \text{OPT}_{c^*}/\text{OPT} \leq 1/\ell$. Similarly, $\text{OPT} \leq \ell \cdot \text{OPT}_{\hat{c}}$, or $\text{OPT}_{\hat{c}}/\text{OPT} \geq 1/\ell$. Thus, $\beta \geq \text{OPT}_{\hat{c}}/\text{OPT} \geq 1/\ell$.

□

Proof (of Lemma 4). Let C_1 be the subset of colors such that for each $c \in C_1$, the algorithm picks at least $\beta \cdot \text{OPT}/(2\ell)$ edges. Note that the algorithm can terminate in two ways (i) it has already picked at least $(\text{OPT}/2\ell) - 1$ edges, and (ii) all the edges have been exhausted. Note that if (i) happens, then we are done. We prove that (ii) cannot happen without (i). Suppose (ii) happens, but not (i). Let OPT' be the subset of OPT containing edges of colors in $C' = C \setminus C_1$. Recall that G_{i-1} is the input graph to the i -th round and G_i is the output graph for $i \geq 1$. The number of edges chosen in i -th round is at most ℓ . Hence, these edges can be adjacent to at most 2ℓ edges in G_{i-1} . In particular, at most 2ℓ less edges of OPT' are contained in G_i compared to G_{i-1} . It follows that the algorithm can pick at least $\lfloor \text{OPT}'/2\ell \rfloor$ edges of colors in C' . As for each color in C' , less than $\beta \cdot \text{OPT}/(2\ell)$ edges are chosen, the algorithm indeed chooses at least $\lfloor \text{OPT}'/2\ell \rfloor$

edges of these colors. The total number of edges chosen by the algorithm is,

$$\begin{aligned}
\sum_{c \in C_1} |M_c| + \sum_{c \in C'} |M_c| &\geq \sum_{c \in C_1} \beta \cdot \text{OPT}/(2\ell) + \lfloor \text{OPT}'/2\ell \rfloor \\
&\geq \sum_{c \in C_1} \text{OPT}_c/2\ell + \lfloor \text{OPT}'/2\ell \rfloor \\
&\geq (\text{OPT}/2\ell) - 1
\end{aligned}$$

But, this is a contradiction to our assumption, and hence the lemma follows.

□