

Online Algorithms for Matchings with Proportional Fairness Constraints and Diversity Constraints

Anand Louis^{a,*}, Meghana Nasre^{b,**}, Prajakta Nimbhorkar^{c,***} and Govind S. Sankar^{d,****}

^aIndian Institute of Science, Bangalore

^bIndian Institute of Technology Madras, Chennai

^cChennai Mathematical Institute, and UMI ReLaX

^dDuke University

ORCID ID: Anand Louis <https://orcid.org/0000-0002-4727-9219>,

Meghana Nasre <https://orcid.org/0000-0003-0290-4444>,

Prajakta Nimbhorkar <https://orcid.org/0000-0002-7601-9555>, Govind S.

Sankar <https://orcid.org/0000-0002-7443-9599>

Abstract. Matching problems with group-fairness constraints and diversity constraints have numerous applications such as in allocation problems, committee selection, school choice, etc. Moreover, online matching problems have lots of applications in ad allocations and other e-commerce problems like product recommendation in digital marketing.

We study two problems involving assigning *items* to *platforms*, where items belong to various *groups* depending on their attributes; the set of items are available offline and the platforms arrive online. In the first problem, we study online matchings with *proportional fairness constraints*. Here, each platform on arrival should either be assigned a set of items in which the fraction of items from each group is within specified bounds or be assigned no items; the goal is to assign items to platforms in order to maximize the number of items assigned to platforms.

In the second problem, we study online matchings with *diversity constraints*, i.e. for each platform, absolute lower bounds are specified for each group. Each platform on arrival should either be assigned a set of items that satisfy these bounds or be assigned no items; the goal is to maximize the set of platforms that get matched. We study approximation algorithms and hardness results for these problems. The technical core of our proofs is a new connection between these problems and the problem of matchings in hypergraphs.

Our experimental evaluation shows the performance of our algorithms on real-world and synthetic datasets exceeds our theoretical guarantees.

1 Introduction

Matchings in graphs is an important problem in both theory and practice, and has received a lot of attention in literature over several years. Computing a maximum matching in a bipartite graph under various constraints is the core of many allocation applications like scheduling [36], school choice [2], ad-auctions [37, 38], resource allocation

[28], healthcare rationing [4] etc. The terminology in these papers varies, and we refer to the two parts of the underlying bipartite graph with the terms *items* and *platforms*. In real-world applications, items may be classified into different groups based on various properties they possess. Seeking to optimize just the cost or utility of an allocation can be unfair to certain groups or may not serve the intended purpose. For instance, while forming committees from a pool of candidates, it is necessary that each committee contains candidates with expertise from all the relevant areas. This is the case, for example, while forming program committees of conferences, teams to work on projects, or expert committees to evaluate project proposals where it is required that a minimum number of members are picked from each of the relevant sub-areas. Additionally, each committee may have a limit on the maximum number of members it can accommodate from a particular sub-area.

In this paper, we consider the scenario where *items* need to be assigned to *platforms*, items are classified into groups, and the platforms have certain *fairness constraints* on the set of items that gets assigned to them from each group. The fairness constraints can be specified in terms of lower bounds on the number items that get assigned to a platform from each group. However, in many applications, stating the fairness constraints in terms of absolute lower bounds is inadequate. For example, in case of school choice, the total number of applications may not be the same each year (see e.g. [17]). Thus, some schools may not fill up all their seats, and specifying constraints on the number of students with each race, ethnicity, and economic background in terms of absolute values do not achieve the desired balance. Similarly, while selecting committees, the number of available candidates and their backgrounds may vary each year. To address this, we consider the notion of *proportional fairness*. Formally, our model and the problem definition are as follows:

Our model: The input instance consists of a bipartite graph $G = (A \cup P, E)$ where A is the set of items and P is the set of platforms, and $(a, p) \in E$ if and only if item a can be assigned to platform p . Let $N(p_j)$ denote the set of items adjacent to p_j in G . Moreover, depending on the properties or attributes they possess, the items are classified into m groups C_1, \dots, C_m where each $C_i \subseteq A$.

An assignment or matching M of items to platforms is a subset of

* Email: anandl@iisc.ac.in

** Email: meghana@cse.iitm.ac.in

*** Corresponding Author. Email: prajakta@cmi.ac.in

**** Email: govind.subash.sankar@duke.edu

E . For a platform p_j , define $M_j = \{a \in A \mid (a, p_j) \in M\}$. Thus M_j denotes the set of items assigned to p_j in M .

We define two problems on this model, depending on the way fairness constraints are specified. The following problem has *proportional fairness constraints*:

Proportionally fair matching problem: The input instance is as described above. Every platform $p_j \in P$ has a lower bound ℓ_j and upper bound u_j , respectively denoting the minimum and maximum number of items that can be assigned to p_j . Further, each platform p_j has associated balance parameters $\alpha_j^{(i)}, \beta_j^{(i)}$ for each group C_i . It is essential that, when $N(p_j) \cap C_i = \emptyset$, $\alpha_j^{(i)}, \beta_j^{(i)} = 0$. A platform p_j is said to be *satisfied* by M if the following holds:

$$\begin{aligned} \ell_j \leq |M_j| &\leq u_j \text{ and for every } i \text{ s.t. } C_i \cap N(p_j) \neq \emptyset \\ \alpha_j^{(i)} |M_j| &\leq |M_j \cap C_i| \leq \beta_j^{(i)} |M_j| \end{aligned}$$

The goal is to compute an assignment M of items to platforms such that the number of items assigned to *satisfied* platforms is maximized.

Additionally, we also consider the problem where the fairness constraints are specified in terms of absolute lower bounds, and the goal is to maximize the number of platforms whose lower bounds are met. We refer to this as the *diverse matching* problem. The objective of maximizing the number of satisfied platforms is motivated by its real-world applications like committee selection, where a committee cannot be set up unless all the constraints are met, or setting up teams to work on projects, as experts from all the areas are necessary for the completion of a project. The objective of maximizing the number of committees or teams formed is a natural one here. The problem is formally defined below.

Diverse matching problem: The input instance is the same as described previously. Each platform specifies a lower bound $\ell_j^{(i)}$ on the number of items it needs from the groups C_i . It is essential that $\ell_j^{(i)} \leq |N(p_j) \cap C_i|$. A platform p_j is said to be satisfied if, for the set of items M_j assigned to it, the following holds:

$$|M_j| \geq \ell_j, \quad \text{and} \quad \forall i, |M_j \cap C_i| \geq \ell_j^{(i)}.$$

The goal is to compute an assignment of items to platforms such that the number of *satisfied* platforms is maximized.

The online setting: In many practical applications like ad-allocation, selecting teams from available pool of candidates, assigning item reviews to customers, ride sharing etc., all the platforms may not be known in advance. Our algorithms have the added advantage that they work in this setting where platforms arrive online over time and items are known in advance.

Even though our problem formulations are of immense importance as exemplified by the numerous aforementioned special cases they generalize, there doesn't seem to be any prior work studying these problems at the level of generality of our formulations; see Section 1.3 for brief summary of the related work.

1.1 Our Contributions

We give hardness and approximation algorithms for the *Proportionally Fair Matching* and *Diverse Matching* problems. Recall that ℓ_j is the minimum number of items required to be assigned to a platform p_j , and define $\ell = \max_j \ell_j$ taken over all platforms. We state our results for the online setting, clearly they also apply to the *offline* setting, where the platforms and items both are known upfront.

The main result of our paper is as follows:

Theorem 1. *There is an $O(n^2)$ -time online algorithm that outputs an assignment M of items to platforms such that the number of items that get assigned in M is a $2(\ell + 1)$ -approximation to the optimum solution of the Proportionally Fair Matching problem, and the fairness constraint for any platform p_j may be violated by at most a fraction $O(\frac{1}{\ell_j})$.*

Formally, for a group C_i and a platform p_j we have

$$|M_j| \left(\alpha_j^{(i)} - \frac{3}{\ell_j} \right) \leq |M_j \cap C_i| \leq |M_j| \left(\beta_j^{(i)} + \frac{3}{\ell_j} \right)$$

Here n is the number of vertices in the underlying bipartite graph G .

The theorem is proved in Section 2.

We give a slightly better approximation guarantee for the *Diverse Matching* problem, without any violation of constraints, even when an item can belong to multiple groups. Following is our result for the *Diverse Matching* problem, proved in Section 3.

Theorem 2. *There is an online algorithm with competitive ratio $(\ell + 1)$ for the Diverse Matching problem where $\ell = \max_{p_j} (\max(\ell_j, \sum_i \ell_j^{(i)}))$.*

We show that the approximation ratios in Theorems 1 and 2 are almost tight by showing that it is NP-Hard to get an approximation ratio larger than $O(\frac{\ell}{\ln \ell})$. Theorem 3 gives hardness of approximation for the *Proportionally Fair Matching* and *Diverse Matching* problems. The proof is given in Section 4.

Theorem 3. *The Proportionally Fair Matching and the Diverse Matching problems are NP-hard to approximate in polynomial time to within a factor of $O(\frac{\ell}{\ln \ell})$ where $\ell = \max_{p_j} \ell_j$. The hardness result holds even when there is a trivial group containing all the items.*

Experimental evaluation: An experimental evaluation of our algorithms shows that their performance on real-world and synthetic datasets significantly exceeds our theoretical guarantees. Our synthetic datasets are generated using a model that loosely resembles a random graph generated from an Erdős-Rényi model, and show that our algorithms perform very well even for small values of average degree of items. This performance is explained by the high theoretical guarantees obtained in Theorem 4 above. In particular, our algorithm for the *Proportionally Fair Matching* problem outperforms the optimal solution in terms of size, owing to the allowed violation in fairness constraints.

Motivated by the above experimental results, we analyze our algorithm for the *Diverse Matching* problem on *Erdős-Rényi random graphs*. This is similar in spirit to the work of [21], where they show that the greedy matching algorithm obtains an almost-optimal matching on Erdős-Rényi random graphs, which are used as one of the models for real-world instances. In our theoretical analysis of the algorithm for the *Diverse Matching* problem on an Erdős-Rényi random graph, we see a similar behaviour in the presence of lower bounds. The random graph model involves a bipartite graph $G = (A \cup P, E)$ where for every $a \in A, p \in P$, the edge (a, p) exists in E with probability ρ . We consider a simplistic scenario where there are Δ groups, each item belongs to one of the Δ groups, and the lower bound of each group is ℓ .

Theorem 4. *For any constant $\epsilon > 0$, the greedy algorithm from Theorem 2 achieves a $(1 - \epsilon)$ -approximation with probability $(1 - \frac{1}{n^\epsilon})$ for instances of the Diverse Matching problem where all the lower bounds are fixed to a constant ℓ , and the underlying graph*

is an Erdős-Rényi random graph with an edge probability of $\rho = \Omega\left(\frac{\log n}{n}\right)$.

Here, the constant c depends on ρ and ϵ . The proof appears in the full version of the paper [33].

1.2 Our Techniques

The technical core of our theoretical results is a new connection between our problems and the *hypergraph matching problem* defined below.

Definition 1 (Hypergraph Matching). Given a k -uniform hypergraph $H = (V, E)$, find the largest matching, viz. a subset of edges that do not intersect.

In this paper, we use the folklore greedy algorithm for hypergraph matching which involves repeatedly choosing a hyperedge that is disjoint from the edges already included in the matching, and adding it to the matching.

Proposition 1. *The greedy algorithm achieves an approximation ratio of Δ on hypergraphs with hyperedges of size $\leq \Delta$.*

1.3 Related Work

The problem of matchings with fairness constraints has been well-studied in recent years and the importance of fairness constraints has been highlighted in literature e.g. [42, 34, 19, 14, 32, 16, 12]. A lot of work in literature has focused on group fairness constraints, modeled as upper bounds on the number of items from each group that can be allocated to a platform. This is referred to as *restricted dominance* in literature [10]. Constraints that are specified in terms of lower bounds constraints have also been considered for different problems like clustering, minimum cut, knapsack (see e.g. [3, 5, 29] for some recent results). Bera et. al [10] consider *proportional fairness constraints* for clustering problems, and give an algorithm with additive violation of constraints. In this work, we consider them in the context of matchings. In [41] the authors consider upper bounds for problems similar to ours. However, the techniques required for the setting with lower bounds in our work are different from the ones used in [41].

There has been more recent work where different models of fairness in matchings have been considered. In [26], individual fairness is addressed, in [7], two-sided fairness is considered in terms of utilities, whereas in [35], group-fairness in terms of the minimum service rate across all groups is studied. In [24], the authors consider group and individual Rawlsian fairness criteria in an online setting, where fairness is attained at the cost of a drop in operator's profit. In [22], group fairness constraints have been considered for the *entity-resolution* problem.

Proportional fairness has been considered for the *candidate selection problem* in [8]. In their setting, the input consists of n candidates and m properties. Each candidate can possess a subset of the m properties and a property p_j is associated with proportional fairness constraints α_j, β_j . Additionally, the input contains a threshold k . The goal is to output a set of at most k candidates which satisfies the proportionality constraints. The authors show that even deciding whether there exists a non-empty feasible set of candidates is NP-hard. They complement the hardness by providing polynomial-time approximation algorithms with a slight violation of the proportional fairness constraints.

Our proportionally fair matching problem can be viewed as selecting multiple committees instead of a single committee / subset. Thus their hardness and inapproximability hold for our problem as well. We therefore consider a special case where every candidate belongs to exactly one group. In this case, if we are interested in selecting a single committee, the problem becomes tractable, however, it remains NP-hard when there are multiple committees to be selected (Theorem 3).

An advantage of our algorithms is that they simply follow the greedy paradigm, and work in online setting as well. On the contrary, those of Bei et al. [8] involve solving an LP or ILP.

In a recent work, [6] consider the proportional fairness model in case of non-bipartite graphs. Their model involves colors on edges, and the proportional fairness constraints are stated in terms of two parameters α, β . The goal is to construct a matching that has at least α and at most β fraction of edges of each color. The running time of their algorithm is exponential in the number of colors, and the violation of proportional fairness parameters also depends on the number of colors. In another work, [39], the authors consider fairness constraints in terms of lower and upper bounds for each group, and additionally consider *individual fairness constraints* denoting the bounds on probabilities with which an item must be matched to a subset of platforms. They output a distribution on group fair matchings so that a randomly picked matching satisfies the individual fairness constraints.

We note that the term *proportional fairness* has also been used in literature in a way different from ours. For instance, in [43], online matchings for ride-hailing platforms have been considered, and the term proportional fairness is used to indicate that the utility that each agent gets should be proportional to the time that the agent spends on the ride-hailing platform. Also, the term *diversity* has been used in various ways in literature (see e.g. [13, 25]). In [9], both sides of the bipartition belong to various groups, referred to as *types* and *blocks*. They give a max-utility assignment that satisfies all the fairness constraints, given only in terms of upper bounds.

The online model for the bipartite matching problem was studied in [31] and they gave a randomized algorithm with a competitive ratio $1 - 1/e$. This was later generalized to online bipartite matchings with concave returns in [18]. In [20], fairness and diversity in online bipartite matchings has been achieved via submodular weight function.

Apart from matchings, fairness constraints have been considered in clustering problems [23], where fair clustering has been reduced to assignment problems and bicriteria approximations are shown [11, 10].

2 The Proportionally Fair Matching Problem

We discuss the Proportionally Fair Matching problem in this section. Here, we want to maximize the number of items matched to those platforms whose proportional fairness constraints are met.

2.1 Algorithm with no violation of constraints

Given an instance G of the *Proportionally Fair Matching* problem, the algorithm involves constructing an instance of the hypergraph matching problem on a hypergraph H with vertex set V and edge set F as follows: $V = \{v_t \mid a_t \in A\} \cup \{p_j \mid p_j \in P\}$ Thus, corresponding to an item $a_t \in A$, we add a vertex v_t to the set V . For every platform p_j we add a vertex p_j to V . We denote by p_j for a ver-

tex in the hypergraph corresponding to the platform p_j . Furthermore, we let $C_j^{(i)} = C_i \cap N(p_j)$.

The edge set is $F = \{\{p_j\} \cup S \mid S \subseteq N(p_j), \ell_j \leq |S| \leq u_j \ \forall i \in [m] \ \alpha_j^{(i)} |S| \leq |S \cap C_j^{(i)}| \leq \beta_j^{(i)} |S|\}$

Thus, there is a hyperedge in H corresponding to each possible assignment of items to p_j that meets the proportional fairness constraints of p_j and all its groups. Further, by the above construction, observe that any assignment that satisfies the platform must correspond to a hyperedge. The size of the largest hyperedge in H is u where $u = \max_j u_j$.

By Proposition 1, this naive reduction to hypergraph matching described above leads to an approximation ratio of $O(u)$ without violating any of the proportional fairness constraints. Given the hardness of approximation for hypergraph matching we do not expect to improve the approximation guarantee using this reduction.

2.2 Improved approximation with violation of constraints:

We would like an approximation factor that depends on ℓ_j instead of u_j since the former is typically much smaller than the latter e.g. in student course allocation, the number of students required to offer a course is typically small whereas the maximum capacity of the course may be much larger. In order to get an improved approximation factor, we pay a price in terms of a slight violation of the fairness constraints. In the proof of Theorem 1 below, we construct a hypergraph H such that the hyperedges corresponding to a platform p_j have size ℓ_j , thereby resulting in an approximation factor $O(\ell)$ where $\ell = \max_j \ell_j$.

Proof of Theorem 1. We reduce the given instance G to a hypergraph matching problem on a suitably constructed hypergraph H . We then use the greedy algorithm to compute a maximal matching M_H in H . However, owing to rounding errors, the corresponding matching M in G may not exactly satisfy the constraints.

Construction of the hypergraph H : Recall that in this problem, we only consider the case where each item belongs to exactly one group per platform. For every item a_i , we create a vertex v_i in H . For every platform p_j , we create t vertices $u_j^{(1)}, \dots, u_j^{(t)}$ where $t = \lfloor \frac{u_j}{\ell_j} \rfloor$. We add the following hyperedges for platform p_j .

$$\left\{ \{u_j^{(k)}\} \cup S \mid 1 \leq k \leq t, S \subseteq N(p_j), |S| = \ell_j, S \text{ satisfies Eqn 1} \right\}.$$

$$\alpha_j^{(i)} \ell_j - 3 \leq |S \cap C_j^{(i)}| \leq \beta_j^{(i)} \ell_j + 3. \quad (1)$$

We need to find a set S satisfying the above property. For this, we keep adding items from a group $C_j^{(i)}$ to S until $|S \cap C_j^{(i)}| \geq \alpha_j^{(i)} \ell_j - 3$. We repeat this for all the groups. To ensure $|S| = \ell_j$, we add items arbitrarily so that $|S \cap C_j^{(i)}| \leq \beta_j^{(i)} \ell_j + 3$ for each group $C_j^{(i)}$. It is easy to see that if such a set S exists, the above process must find it, and the disjointness of groups is crucial here.

Improving the running time to $O(n^2)$: The construction of H needs time n^ℓ . However, we improve it to $O(n^2)$ as follows. Instead of explicitly constructing all the hyperedges of H , we create a collection of disjoint hyperedges by constructing hyperedges one by one, ensuring that the hyperedge being created is disjoint from the previously created ones. This results in a simple greedy matching M_H of H . Algorithm 1 shows this directly for the instance G .

Algorithm 1 Improved algorithm for Proportionally Fair Matching

- 1: **for** every platform p_j that arrives online **do**
 - 2: **while** $\exists S \subseteq N(p_j)$ of size ℓ_j satisfying Equation 1 **do**
 - 3: Match all items from S to p_j and remove them.
 - 4: **end while**
 - 5: **end for**
-

Let the matching M_H contain $t' \leq t$ hyperedges corresponding to platform p_j , each containing a distinct vertex from $u_j^{(1)}, \dots, u_j^{(t)}$. Note that the items in these hyperedges are assigned to p_j in the corresponding matching M in G . Let this set be M_j . By the construction of H , we observe that if $t' \geq 1$, we have $\ell_j \leq |M_j| \leq u_j$. Moreover since every hyperedge satisfies Eq. 1, we can add Equation 1 across all the t' hyperedges to get

$$\alpha_j^{(i)} |M_j| - 3t' \leq |M_j \cap C_j^{(i)}| \leq \beta_j^{(i)} |M_j| + 3t' \\ \left(\alpha_j^{(i)} - \frac{3}{\ell_j} \right) |M_j| \leq |M_j \cap C_j^{(i)}| \leq \left(\beta_j^{(i)} + \frac{3}{\ell_j} \right) |M_j|.$$

Thus, we violate the constraints by at most a factor of $\frac{3}{\ell_j}$.

We show the desired approximation ratio of $2(\ell + 1)$ as follows. In Lemma 2 below, we show that for any optimum matching M_{OPT} in G , there exists a matching S_H in G such that $|S_H| \geq \frac{|M_{OPT}|}{2}$. Then we give a transformation to get a matching S_H in H . The $2(\ell + 1)$ approximation then follows from Proposition 1. The final solution satisfies the relaxed fairness constraints in Equation 1. \square

Lemma 2. *The hypergraph H contains a matching S_H such that the number of items covered by hyperedges in S_H is at least half the value of the optimum of original instance of Proportionally Fair Matching and violates the fairness constraints by at most a factor of $\frac{3}{\ell_j}$ for each platform p_j .*

Proof. Let M_{OPT} be the optimum matching in G . We focus our attention on some platform p_j . Let it be matched to a set $M_{OPT,j}$ of items in M_{OPT} . Let $|M_{OPT,j}| = q\ell_j + r$ for non-negative integers q, r where $r < \ell_j$ and $q > 0$. We construct a solution S whose restriction to the neighbours of p_j , S_j , satisfies $|S_j| = q\ell_j$ matched items. Further, for each group $C_j^{(i)}$ of p_j , $S_j^{(i)}$ satisfies the following slightly relaxed constraint. For each i , let $M_{OPT,j}^{(i)} = M_{OPT,j} \cap C_j^{(i)}$. Then S_j needs to satisfy $|S_j| = q\ell_j$ and

$$|S_j^{(i)}| = \left\lceil \frac{q\ell_j}{q\ell_j + r} |M_{OPT,j}^{(i)}| \right\rceil \quad \text{or} \quad |S_j^{(i)}| = \left\lfloor \frac{q\ell_j}{q\ell_j + r} |M_{OPT,j}^{(i)}| \right\rfloor.$$

This can easily be done by removing vertices one by one from $M_{OPT,j}^{(i)}$ for various i . Then we have

$$\frac{q\ell_j}{q\ell_j + r} |M_{OPT,j}^{(i)}| - 1 \leq |S_j^{(i)}| \leq \frac{q\ell_j}{q\ell_j + r} |M_{OPT,j}^{(i)}| + 1 \\ \implies \alpha_j^{(i)} q\ell_j - 1 \leq |S_j^{(i)}| \leq \beta_j^{(i)} q\ell_j + 1. \quad (2)$$

Thus, there is a solution of size $q\ell_j$ that violates the lower and upper bounds by at most 1. Note that we can repeat this for each satisfied platform, and this process may possibly reduce the size of the matching by a factor of 2, i.e., $|S| \geq \frac{1}{2} |M_{OPT}|$. This is because, the loss is $r \leq \frac{q\ell_j + r}{2}$ because $r < \ell_j$ and $q > 0$.

Now, we want to construct a matching S_H in H from the solution S . To do this, we divide the above $q\ell_j$ items in S_j into q hyperedges

of size ℓ_j each such that each hyperedge satisfies Equation 1. These hyperedges together form the matching \mathcal{S}_H in H . Through the arguments given below, we show that this is possible.

We do this by setting up a bipartite matching instance between items and hyperedges. We create a vertex for each item and hyperedge. Consider the x items from some group $C_j^{(i)}$ that we need to assign to q hyperedges. We fractionally match x_i/q items from that group to each hyperedge.

Observe that this can be done in a way so that each hyperedge has at most two fractional items matched to it from one group: We imagine this as dividing the real interval between $[0, x]$ into q divisions. Each division can then be viewed as the union of a set of ‘integral’ intervals of the form $[z, z + 1]$ for some integer z along with at most two ‘fractional’ intervals (corresponding to items being fractionally being matched). We then repeat this for all the groups. For example, to assign 7 items $\{i_1, \dots, i_7\}$ to 3 hyperedges f_1, f_2, f_3 , we assign i_1, i_2 and $\frac{1}{3}$ of i_3 to f_1 , the remaining $\frac{2}{3}$ of i_3 , whole i_4 , and $\frac{2}{3}$ of i_5 to f_2 , and the remaining $\frac{1}{3}$ of i_5 and whole of i_6, i_7 to f_3 .

Now, we consider the fractional bipartite matching between the items and hyperedges. The edges have weight corresponding to the fraction of the item that was assigned. Observe that this graph admits a fractional matching with the property that on the item side, the weights of edges adjacent to a vertex sum up to 1 and on the side of hyperedges, the weights of the adjacent edges of every vertex sum up to ℓ_j . From the integrality of the bipartite matching polytope [40], there is an underlying integral matching such that every item is assigned to exactly 1 hyperedge and every hyperedge is assigned exactly ℓ_j items. We create our hyperedges based on this matching.

Since there are only two fractional items from a group with an edge to a hyperedge, it can only violate the constraints by at most an additive factor of 2. Thus, we end up with hyperedges S that satisfy the constraint

$$\begin{aligned} \frac{|S_j^{(i)}|}{q} - 2 &\leq |S \cap C_j^{(i)}| \leq \frac{|S_j^{(i)}|}{q} + 2 \\ \implies \alpha_j^{(i)} \ell_j - \frac{1}{q} - 2 &\leq |S \cap C_j^{(i)}| \leq \beta_j^{(i)} \ell_j + \frac{1}{q} + 2. \end{aligned}$$

The hyperedges here, with addition of a distinct $u_j^{(i)}$ vertex, $1 \leq i \leq q$, form a disjoint collection of hyperedges in H , thus giving the matching \mathcal{S}_H in H . Since our algorithm will not know the value of q beforehand, we choose the worst case of $q = 1$. Thus, the hyperedges satisfy

$$\alpha_j^{(i)} \ell_j - 3 \leq |Z \cap C_j^{(i)}| \leq \beta_j^{(i)} \ell_j + 3. \quad \square$$

We note that the high-level idea of breaking a problem with group-fairness constraints into ‘smaller’ problems has been studied in the context of other problems as well such as fair clustering [15], fair rankings [27] etc. However, doing so is problem specific and there is no known generic way of doing this for any problem.

3 Matchings with Diversity Constraints

We give a reduction from the *Diverse Matching* problem to the *Hypergraph Matching Problem* which implies an approximation algorithm for the *Diverse Matching* problem, and thereby prove Theorem 2. As in Section 2, we use the notation $C_j^{(i)} = N(p_j) \cap C_i$.

Algorithm 2 Algorithm for *Diverse Matching*

- 1: **for** platform p_j that arrives online **do**
- 2: Greedily construct a set $S \subseteq N(p_j)$ such that

$$\ell_j \leq |S| \leq \max(\ell_j, \sum_i \ell_j^{(i)}) \text{ and } |S \cap C_j^{(i)}| \geq \ell_j^{(i)} \quad \forall i \in [m]$$

- 3: Match all items from S to p_j and remove them.
 - 4: **end for**
-

Proof of Theorem 2. Given an instance of *Diverse Matching* on a bipartite graph G with parts (A, P) , we construct an instance of the hypergraph matching problem on a hypergraph H with vertex set V . We add one vertex v_i in V corresponding to every item $a_i \in A$, and a new vertex u_j in V for each platform $p_j \in P$. Let $N(p_j)$ be the neighbourhood of platform p_j in G . Then we add the following hyperedges to the graph

$$\left\{ \{u_j\} \cup S \mid S \subseteq N(p_j), \ell_j \leq |S| \leq \max(\ell_j, \sum_k \ell_j^{(k)}) \right. \\ \left. \forall k, |S \cap C_j^{(k)}| \geq \ell_j^{(k)} \right\}.$$

Thus, there is a hyperedge in H corresponding to each possible assignment of items to p_j that satisfies the lower bounds of p_j and all its groups. Observe that the largest hyperedge in the resulting hypergraph has size $\ell + 1$. It is immediate that every hyperedge in the matching corresponds to an assignment of items to a platform that satisfies the platform. Further, observe that any assignment that satisfies the platform must correspond to a hyperedge.

As stated earlier, we can now use the hypergraph matching greedy algorithm. Observe that the number of hyperedges can be as large as $O(n^\ell)$ and hence the time complexity would be as high too. However, we achieve $O(n^2)$ time complexity as follows. Instead of explicitly constructing all the hyperedges, we keep constructing and picking one arbitrary hyperedge at a time that is disjoint from the previous ones. This is given in Algorithm 2 in terms of the setting in *Diverse Matching*. Thus for each platform p_j , we keep adding items from each group to a set S_j until the lower bound of the group is met by the items in S_j . Once the lower bounds of all the groups are met, if $|S_j| < \ell_j$, we add items arbitrarily to S_j until $|S_j| = \ell_j$. We ensure that an item is picked at most once. In the worst case, $|S_j| = \max(\ell_j, \sum_k \ell_j^{(k)})$. The theorem then follows via Proposition 1. \square

4 Hardness Results

We prove Theorem 3 by giving a reduction from the *hypergraph matching* problem to *Diverse Matching*. The hardness of approximation for hypergraph matching, shown in [30], then implies that the *Diverse Matching* is NP-hard to approximate within a factor of $\frac{\ell}{\ln \ell}$. This hardness result holds even when there are no groups, and all the platforms have the same lower bound ℓ .

Proof of Theorem 3. The hardness reductions are described below.

Reduction for the *Diverse Matching* problem: The reduction involves constructing an instance of the *Diverse Matching* problem from a given hypergraph matching instance $H = (F, V)$, which is a k -uniform hypergraph. We construct a bipartite graph G with parts (A, P) as follows: Set $\ell = k$. The set A contains an item a_i for every vertex $v_i \in V$, the set P contains a platform p_j for every $e_j \in F$,

and the edges of G are all pairs (a_i, p_j) such that $v_i \in e_j$. The lower bound of p_j is set to the size of e_j i.e. ℓ , and we have no groups in this case.

Now we show that, for every matching M in H of size r , there is an assignment of items to platforms that satisfies the lower bounds of r platforms and vice versa. Suppose there is a matching M in H of size r . Then for every hyperedge $e_j \in M$, we assign items $\{a_i \mid v_i \in e_j\}$ to the corresponding platform p_j . This meets the lower bound of p_j , since the size of e_j is equal to the lower bound of p_j for each j . Thus, since every e_j has exactly k vertices, and the edges in M do not share vertices, we get an assignment of items to platforms that satisfies the lower bounds of r platforms. Similarly, suppose there is an assignment of items to platforms that satisfies the lower bounds of r platforms. Hence the assignment results in r platforms that are satisfied. Let p_j be some platform that is satisfied. Then we choose the corresponding hyperedge e_j into the solution. Since each item is matched to exactly one platform, corresponding hyperedges in the solution are disjoint. This gives us a matching in the hypergraph, thereby establishing a one-to-one correspondence between matchings in H and the matchings in G satisfying all the lower bounds.

Reduction for the Proportionally Fair Matching problem: The above reduction also gives a hardness for the *Proportionally Fair Matching* problem as follows. After constructing the bipartite graph G as above, for each platform p_j , set $u_j = \ell_j = k$. Further, set the number of groups to be 1. Thus we have $C_j^{(1)} = N(p_j)$ for each platform p_j . Also, set $\alpha_j^{(1)} = 0, \beta_j^{(1)} = 1$. It can be easily seen that the number of items matched to satisfied platforms is precisely k times the number of satisfied platforms. The one-to-one correspondence between the set of matchings in H and the set of matchings in G that satisfy the proportional fairness constraints can be established as in the case of the *Diverse Matching* problem. \square

5 Experiments

We present the results of our experiments for Algorithm 1 and Algorithm 2 for the *Proportionally Fair Matching* and the *Diverse Matching* problems respectively. We evaluate the algorithms on two kinds of datasets. The first one is a smaller dataset containing anonymized data of the course allocation process at IIT Madras, labelled Real-1 through Real-3. The second is synthetic data generated using a random process resembling a random graph generated from an Erdős-Rényi model. Our code and dataset are available at [1].

Data Sets and Setup: Each of the three real-world datasets (Real-1, Real-2, Real-3), have around 3000 students and 100 courses. Each course has a lower quota of 5, denoting the minimum number of students needed to operate the course. As an instance of *Diverse Matching*, we would like to maximize the number of courses that satisfy this requirement. The courses are all from an elective category, so each student is assigned only one course. Students are partitioned into groups based on their majors, and there are 5 groups overall. The synthetic datasets contain 250 courses and 10,000 students. These were generated as follows. For every student, a degree was chosen uniformly at random between 1 and an input parameter. All experiments were done on a desktop running 64-bit Windows using a 3.6 GHz Intel i7-7700 processor with 32.0 GB RAM.

In our experiments for both problems our algorithm is denoted as ALG1 in the respective sections. We use two standard heuristics that improve the performance in practice, though they do not improve the worst-case theoretical guarantees because of the hardness results. The first heuristic (denoted as ALG2) is to prioritize matching the

lowest-degree item to a platform when considering its neighbors. The second one (denoted as ALG3) is to use augmenting paths when a platform is left with no unmatched items among its neighbors. We compare the optimum value obtained via solving an ILP with the output of our algorithm. We present our ILPs for the two problems in the sections below.

5.1 Proportionally Fair Matching

We use the following ILP for the proportionally fair matching problem: The input bipartite graph is $G = (A \cup P, E)$

$$\begin{aligned} \text{Maximize } & \sum_{(a,p) \in E} x_{(a,p)} && \text{subject to} \\ & \alpha_p^{(k)} \cdot \sum_{(a,p) \in E} x_{(a,p)} \leq \sum_{(a,p) \in E, a \in C_p^{(k)}} x_{(a,p)} && (3) \\ & \sum_{(a,p) \in E, a \in C_p^{(k)}} x_{(a,p)} \leq \beta_p^{(k)} \cdot \sum_{(a,p) \in E} x_{(a,p)} && (4) \\ & \sum_{(a,p) \in E} x_{(a,p)} \leq 1 \text{ for each } a \in A && (5) \\ & x_{(a,p)} \in \{0, 1\} && \text{for each } (a, p) \in E \end{aligned}$$

The proportional fairness constraints (3) and (4) are for every group $C_p^{(k)}$ for each platform p whereas the constraint (5) encodes that each applicant is matched to at most one platform.¹ We run our experiments for *Proportionally Fair Matching* on a smaller dataset owing to a limitation in computational resources. We have 100 courses and 2000 students. Every course has the same set of groups. There are 20 groups, each containing 100 students. Every course has a lower bound of 10 students overall. Every course p_j and every group k have $\alpha_j^{(k)} = 0.025$ and $\beta_j^{(k)} = 0.1$. We note that our algorithms solve a relaxed instance. This explains the strange phenomenon that our algorithm outperforms the OPT in some instances. The OPT is the optimal solution to a more constrained instance.

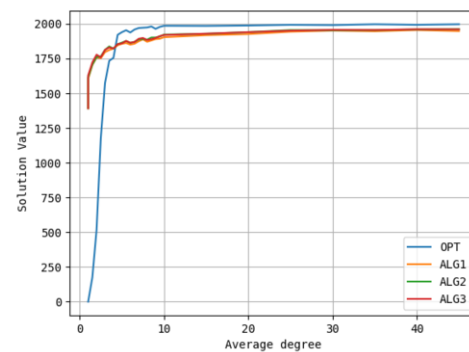


Figure 1. *Proportionally Fair Matching*: Solution value vs average degree of the random graph.

5.2 Diverse Matching

Following is the ILP for the diverse matching problem:

¹ This ILP can be found in the file named ILP_student_max.py in our submitted source code.

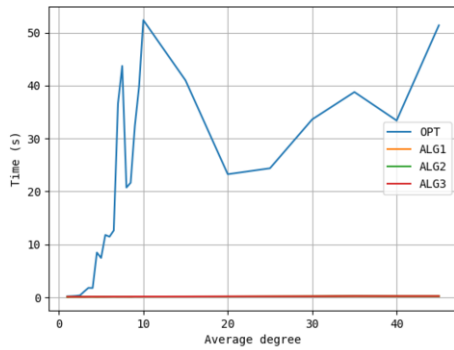


Figure 2. *Proportionally Fair Matching*: Running time vs average degree of the random graph.

$$\begin{aligned}
 & \text{Maximize } \sum_{p \in P} y_p && \text{subject to} \\
 & \sum_{(a,p) \in E} x_{(a,p)} \geq \ell_p \cdot y_p && \text{for each } p \in P \\
 & \sum_{(a,p) \in E, a \in C_p^{(k)}} x_{(a,p)} \geq \ell_p^{(k)} \cdot y_p && \text{for each group } C_p^{(k)} \text{ of } p \\
 & \sum_{(a,p) \in E} x_{(a,p)} \leq 1 && \text{for each } a \in A \\
 & y_p \in \{0, 1\} && \text{for each } p \in P \\
 & x_{(a,p)} \in \{0, 1\} && \text{for each } (a, p) \in E
 \end{aligned}$$

We run our experiments for *Diverse Matching* on real world and synthetically generated instances where the number of courses is 250 and the number of students is 10,000. There are 20 groups, each containing 500 students. Every course has a lower bound of 2 for each group. We vary the average degree of the students from 1 through 125.

All values were averaged over 15 runs. See Table 1 and Table 2 for solution value and running time comparison between our algorithms and OPT.

Dataset	OPT	ALG1	ALG2	ALG3
Real-1	34	29.53	31.73	33.20
Real-2	31	27.73	29.40	30.86
Real 3	31	26.07	28.73	30.00

Table 1. *Diverse Matching*: Comparison of solution values of (ALG1, ALG2, ALG3) and an ILP that finds the Optimum (OPT).

Dataset	OPT1	ALG1	ALG2	ALG3
Real-1	0.37	0.11	0.12	0.12
Real-2	0.34	0.11	0.12	0.12
Real 3	0.42	0.13	0.12	0.12

Table 2. *Diverse Matching*: Runtime (in seconds) comparison between ALG1, ALG2, ALG3 and an ILP that finds the optimum (OPT).

We observe that beyond a small threshold, the algorithm performed almost as well as the ILP. Our algorithm is much faster the ILP, particularly for dense graphs. See Figure 4 for a trend on performance of algorithm vs the average degree.

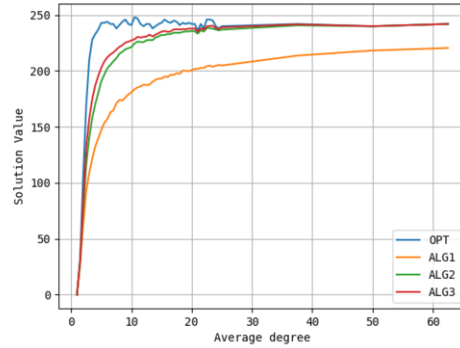


Figure 3. *Diverse Matching*: Solution value vs average degree of the random graph for synthetic datasets.

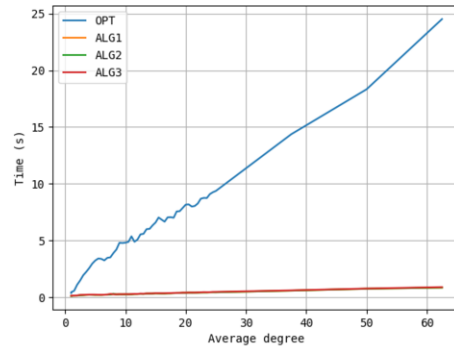


Figure 4. *Diverse Matching*: Running time vs average degree of the random graph for synthetic datasets.

6 Discussion

In this paper, we studied bipartite matching problems with proportional fairness constraints and diversity constraints. To the best of our knowledge, these constraints have not been considered particularly with the objective to maximize the number of platforms whose constraints are satisfied. Our algorithms exploited a connection to the hypergraph matching problem. Our algorithms generalize to the setting where each platform defines its own groups on the set of items.

Our approximation algorithm for the *Proportionally Fair Matching* problem violates the fairness constraints by a small amount; obtaining a polynomial-time $O(\ell)$ -approximation algorithm without violating the fairness constraints is an interesting open question.

References

- [1] Github repository, <https://github.com/prajaktanimbhorkar/ecai23.git>.
- [2] Atila Abdulkadiroglu and T. Sönmez, ‘School choice: A mechanism design approach’, *The American Economic Review*, (2003).
- [3] Anna Arutyunova and Melanie Schmidt, ‘Achieving anonymity via weak lower bound constraints for k-median and k-means’, in *38th International Symposium on Theoretical Aspects of Computer Science, STACS 2021, March 16-19, 2021, Saarbrücken, Germany (Virtual Conference)*, eds., Markus Bläser and Benjamin Monmege, volume 187 of *LIPICs*, pp. 7:1–7:17. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, (2021).
- [4] Haris Aziz and Florian Brandl, ‘Efficient, fair, and incentive-compatible healthcare rationing’, in *EC ’21: The 22nd ACM Conference on Economics and Computation, Budapest, Hungary, July 18-23, 2021*, eds., Péter Biró, Shuchi Chawla, and Federico Echenique, pp. 103–104. ACM, (2021).
- [5] Amy Babay, Michael Dinitz, Prathyush Sambaturu, Aravind Srinivasan, Leonidas Tsepenekas, and Anil Vullikanti, ‘Fair disaster containment via graph-cut problems’, *CoRR*, **abs/2106.05424**, (2021).
- [6] Sayan Bandyapadhyay, Fedor V. Fomin, Tanmay Inamdar, and Kirill Simonov, Proportionally fair matching with multiple groups, 2023.
- [7] Kinjal Basu, Cyrus DiCiccio, Heloise Logan, and Noureddine El Karoui, ‘A framework for fairness in two-sided marketplaces’, *CoRR*, **abs/2006.12756**, (2020).
- [8] Xiaohui Bei, Shengxin Liu, Chung Keung Poon, and Hongao Wang, ‘Candidate selections with proportional fairness constraints’, *Auton. Agents Multi Agent Syst.*, **36**(1), 5, (2022).
- [9] Nawal Benabbou, Mithun Chakraborty, Xuan-Vinh Ho, Jakub Sliwinski, and Yair Zick, ‘Diversity constraints in public housing allocation’, in *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems, AAMAS 2018, Stockholm, Sweden, July 10-15, 2018*, eds., Elisabeth André, Sven Koenig, Mehdi Dastani, and Gita Sukthankar, pp. 973–981. International Foundation for Autonomous Agents and Multiagent Systems Richland, SC, USA / ACM, (2018).
- [10] Suman Bera, Deeparnab Chakraborty, Nicolas Flores, and Maryam Negahbani, ‘Fair algorithms for clustering’, in *Advances in Neural Information Processing Systems*, (2019).
- [11] Ioana Oriana Bercea, Martin Groß, Samir Khuller, Aounon Kumar, Clemens Rösner, Daniel R. Schmidt, and Melanie Schmidt, ‘On the cost of essentially fair clusterings’, in *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, APPROX/RANDOM 2019, September 20-22, 2019, Massachusetts Institute of Technology, Cambridge, MA, USA*, eds., Dimitris Achlioptas and László A. Végh, volume 145 of *LIPICs*, pp. 18:1–18:22. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, (2019).
- [12] Tolga Bolukbasi, Kai-Wei Chang, James Zou, Venkatesh Saligrama, and Adam Kalai, ‘Man is to computer programmer as woman is to homemaker? debiasing word embeddings’, in *NIPS’16*, (2016).
- [13] L. Elisa Celis, Amit Deshpande, Tarun Kathuria, and Nisheeth K. Vishnoi, ‘How to be fair and diverse?’, *CoRR*, **abs/1610.07183**, (2016).
- [14] L. Elisa Celis, Damian Straszak, and Nisheeth K. Vishnoi, ‘Ranking with fairness constraints’, in *ICALP*, (2017).
- [15] Flavio Chierichetti, Ravi Kumar, Silvio Lattanzi, and Sergei Vassilvitskii, ‘Fair clustering through fairlets’, in *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pp. 5029–5037, (2017).
- [16] Matthew Costello, James Hawdon, Thomas Ratliff, and Tyler Grantham, ‘Who views online extremism? individual attributes leading to exposure’, *Comput. Hum. Behav.*, (2016).
- [17] Cowen Institute, ‘Case studies of school choice and open enrollment in four cities’, Technical report. Cowen Institute, (2011).
- [18] Nikhil Devanur and Kamal Jain, ‘Online matching with concave returns’, in *In Proc. STOC 2012*, (May 2012).
- [19] Nikhil R. Devanur, Kamal Jain, and Robert D. Kleinberg, ‘Randomized primal-dual analysis of ranking for online bipartite matching’, in *SODA ’13*, (2013).
- [20] John P. Dickerson, Karthik Abinav Sankararaman, Aravind Srinivasan, and Pan Xu, ‘Balancing relevance and diversity in online bipartite matching via submodularity’, in *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019*, pp. 1877–1884. AAAI Press, (2019).
- [21] Martin Dyer, Alan Frieze, and Boris Pittel, ‘The average performance of the greedy matching algorithm’, *The Annals of Applied Probability*, **3**(2), 526–552, (1993).
- [22] Vasilis Efthymiou, Kostas Stefanidis, Evaggelia Pitoura, and Vassilis Christophides, ‘Fairer: Entity resolution with fairness constraints’, in *CIKM ’21: The 30th ACM International Conference on Information and Knowledge Management, Virtual Event, Queensland, Australia, November 1 - 5, 2021*, eds., Gianluca Demartini, Guido Zuccon, J. Shane Culpepper, Zi Huang, and Hanghang Tong, pp. 3004–3008. ACM, (2021).
- [23] Seyed A. Esmaeili, Brian Brubach, Aravind Srinivasan, and John P. Dickerson, ‘Fair clustering under a bounded cost’, *CoRR*, **abs/2106.07239**, (2021).
- [24] Seyed A. Esmaeili, Sharmila Duppala, Vedant Nanda, Aravind Srinivasan, and John P. Dickerson, Rawlsian fairness in online bipartite matching: Two-sided, group, and individual, 2022.
- [25] Fedor V. Fomin, Petr A. Golovach, Lars Jaffke, Geevarghese Philip, and Danil Sagunov, ‘Diverse pairs of matchings’, in *31st International Symposium on Algorithms and Computation, ISAAC 2020, December 14-18, 2020, Hong Kong, China (Virtual Conference)*, eds., Yixin Cao, Siu-Wing Cheng, and Mingming Li, volume 181 of *LIPICs*, pp. 26:1–26:12. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, (2020).
- [26] David García-Soriano and Francesco Bonchi, ‘Fair-by-design matching’, *Data Min. Knowl. Discov.*, **34**(5), 1291–1335, (2020).
- [27] Sruthi Gorantla, Amit Deshpande, and Anand Louis, ‘On the problem of underranking in group-fair ranking’, in *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 3777–3787. PMLR, (18–24 Jul 2021).
- [28] Hassan Halabian, Ioannis Lambadaris, and Chung-Horng Lung, ‘Optimal server assignment in multi-server parallel queueing systems with random connectivities and random service failures’, in *IEEE International Conference on Communications (ICC)*, (2012).
- [29] Lu Han, Chunlin Hao, Chenchen Wu, and Zhenning Zhang, ‘Approximation algorithms for the lower-bounded knapsack median problem’, in *Algorithmic Aspects in Information and Management*, pp. 119–130. Springer International Publishing, (2020).
- [30] Elad Hazan, Shmuel Safra, and Oded Schwartz, ‘On the complexity of approximating k-dimensional matching’, in *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*, pp. 83–97, Berlin, Heidelberg, (2003). Springer Berlin Heidelberg.
- [31] R. M. Karp, U. V. Vazirani, and V. V. Vazirani, ‘An optimal algorithm for on-line bipartite matching’, *STOC ’90*, (1990).
- [32] Matthew Kay, Cynthia Matuszek, and Sean A. Munson, ‘Unequal representation and gender stereotypes in image search results for occupations’, in *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems, CHI ’15*, (2015).
- [33] Anand Louis, Meghana Nasre, Prajakta Nimbhorkar, and Govind S. Sankar, ‘Online algorithms for matchings with proportional fairness constraints and diversity constraints’, *CoRR*, **abs/2208.11378**, (2023).
- [34] Hanan Luss, ‘On equitable resource allocation problems: A lexicographic minimax approach’, *Operations Research*, (1999).
- [35] Will Ma, Pan Xu, and Yifan Xu, Group-level fairness maximization in online bipartite matching, 2021.
- [36] Nick McKeown, Venkat Anantharam, and Jean Walrand, ‘Achieving 100% throughput in an input-queued switch’, in *INFOCOM ’96*, (1996).
- [37] Aranyak Mehta, ‘Online matching and ad allocation’, *Foundations and Trends® in Theoretical Computer Science*, (2013).
- [38] Aranyak Mehta, Amin Saberi, Umesh Vazirani, and Vijay Vazirani, ‘Adwords and generalized online matching’, *J. ACM*, (2007).
- [39] Atasi Panda, Anand Louis, and Prajakta Nimbhorkar, Bipartite matchings with group fairness and individual fairness constraints, 2022.
- [40] M.D. Plummer and L. Lovász, *Matching Theory*, ISSN, Elsevier Science, 1986.
- [41] Govind S. Sankar, Anand Louis, Meghana Nasre, and Prajakta Nimbhorkar, ‘Matchings with group fairness constraints: Online and offline algorithms’, in *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI 2021, Virtual Event / Montreal, Canada, 19-27 August 2021*, pp. 377–383. ijcai.org, (2021).
- [42] Erel Segal-Halevi and Warut Suksompong, ‘Democratic fair allocation of indivisible goods’, *Artificial Intelligence*, (2019).
- [43] Tom Sühr, Asia J. Biega, Meike Zehlike, Krishna P. Gummadi, and Abhijnan Chakraborty, ‘Two-sided fairness for repeated matchings in two-sided markets: A case study of a ride-hailing platform’, *KDD ’19*, (2019).